



UvA-DARE (Digital Academic Repository)

From simulations to surrogates

Neural networks enhancing burn wound healing predictions

Papapanagiotou, Ioannis; Bumbuc, Roland V.; Korkmaz, H. Ibrahim; Krzhizhanovskaya, Valeria; Sheraton, Vivek M.

DOI

[10.1016/j.jocs.2025.102593](https://doi.org/10.1016/j.jocs.2025.102593)

Publication date

2025

Document Version

Final published version

Published in

Journal of Computational Science

License

CC BY

[Link to publication](#)

Citation for published version (APA):

Papapanagiotou, I., Bumbuc, R. V., Korkmaz, H. I., Krzhizhanovskaya, V., & Sheraton, V. M. (2025). From simulations to surrogates: Neural networks enhancing burn wound healing predictions. *Journal of Computational Science*, 89, Article 102593. <https://doi.org/10.1016/j.jocs.2025.102593>

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).






Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (<https://dare.uva.nl>)



From simulations to surrogates: Neural networks enhancing burn wound healing predictions

Ioannis Papapanagiotou ^a , Roland V. Bumbuc ^{a,b,c,d,e} , H. Ibrahim Korkmaz ^{b,c,d,e,f,g} ,
Valeria Krzhizhanovskaya ^{a,b} , Vivek M. Sheraton ^{a,b} ,^{*}

^a University of Amsterdam, UvA, Amsterdam, Nord-Holland, The Netherlands

^b Computational Science Lab, Informatics Institute, University of Amsterdam, UvA - LAB42, Amsterdam, Nord-Holland, The Netherlands

^c Amsterdam UMC, Location VUmc, Amsterdam, Nord-Holland, The Netherlands

^d Department of Plastic, Reconstructive and Hand Surgery, Amsterdam Movement Sciences(AMS) Institute, Amsterdam UMC, Location VUmc, Amsterdam, Nord-Holland, The Netherlands

^e Department of Molecular Cell Biology and Immunology, Amsterdam Infection and Immunity (AII) Institute, Amsterdam UMC, Location VUmc, Amsterdam, Nord-Holland, The Netherlands

^f Burn Center and Department of Plastic and Reconstructive Surgery, Red Cross Hospital, Beverwijk, Nord-Holland, The Netherlands

^g Burn Research Lab, Alliance of Dutch Burn Care, Beverwijk, Nord-Holland, The Netherlands

ARTICLE INFO

Keywords:

Burns
Immune response
Agent-based model
Long short-term memory neural networks
Physics-informed neural networks

ABSTRACT

Burn injuries trigger substantial inflammation, complicating wound healing and potentially leading to severe systemic complications. Understanding the immune response to burns is crucial for improving treatment. Although agent-based models (ABMs) are valuable for studying these interactions, they are computationally demanding. This paper explores the integration of neural networks (NNs) as surrogate models to approximate and forecast ABM simulation results in predicting cytokine concentrations over time and space. We present the development of a baseline ABM using the CompuCell3D software, simulating the innate immune response and generating extensive cytokine concentration data. This data is processed and prepared for neural network training, involving data cleaning, transformation into suitable formats, and a time-series-aware train-test split. We then implement and assess various neural network architectures. Each model is designed to capture the temporal and spatial dynamics of cytokine concentrations, with adjusted model architectures (kernels, number of layers, neurons per layer) to better suit this problem. The models are evaluated using Mean Squared Error, R-squared, and Mean Absolute Percentage Error. In this paper, we assess how different NN architectures (convolutional neural networks (CNNs), long short-term memory (LSTM) neural networks, attention mechanisms, and physics-informed neural networks (PINNs)) predict the concentration of cytokines in this biological system. We find that STA-LSTM generally performs best across statistical metrics.

1. Introduction

The immune system, a highly organised network of specialised cells and molecules, plays a crucial role in defending the body against infections and other harmful threats, such as wound injury [1]. Burn wound injury triggers a profound inflammatory response that can hinder wound healing and cause systemic complications, including sepsis, pneumonia, and multi-organ failure [2]. Consequently, gaining a deeper understanding of the immune response to burn wounds is crucial to prevent or alleviate these complications.

Computational modelling has become an important tool used to unravel the complexities of the immune response to systemic inflammations [3] and to translate this knowledge into improved clinical

practices [4,5]. Computational modelling offers a systematic approach to simulating biological processes [6–9], allowing researchers to investigate how different factors influence the immune response and the trajectory of wound healing.

Among the various computational modelling approaches [10,11], agent-based modelling (ABM) has gained prominence in this field [12]. ABMs represent individual biological entities as computer agents, allowing for a detailed representation of the interactions and decision-making processes that govern the immune response. This granular level of detail provides valuable insight into the intricate network of immune cells and their interactions, enabling researchers to better

^{*} Correspondence to: University of Amsterdam, UvA - LAB42, The Netherlands.
E-mail address: v.s.muniraj@uva.nl (V.M. Sheraton).

understand how burn injury disrupts these processes and contributes to the development of complications.

Despite their strengths, ABMs can be computationally expensive and time-consuming to run, limiting their application in clinical settings. To address this challenge, researchers investigate the use of neural networks (NNs) as surrogate models for ABMs [13–16] with the task of forecasting and approximating the results of ABM simulations. Neural networks, inspired by the structure and function of the human brain, are capable of learning from data and making predictions. By training neural networks with ABMs, it is possible to develop surrogate models that can accurately simulate the behaviour of the ABM without the computational burden of running the full simulation.

In this paper, we explore the implementation of neural networks as surrogate models for an agent-based model of the immune response to burn wounds. We compare different neural networks tailored for this task to determine the most effective approach. The research focusses on making use of these models to forecast and approximate the ABM's results, with regard to cytokine concentrations on a temporal and spatial two-dimensional scale.

2. Methods

2.1. Baseline agent-based model

Our agent-based model (ABM) is described in detail in [17]. Implemented using the CompuCell3D software framework [18], this model offers a highly detailed, dynamic representation of post-burn inflammation. The solved diffusion equations describe the dynamics of six key solutes/proteins: IL-8, IL-1 β , IL-6, IL-10, TNF- α , and TGF- β 1. Each equation models the temporal change in cytokine concentration and includes a diffusion term (D), one or two secretion terms (K), a decay term (μ), and an endocytosis term (θ) specific to IL-8. The PDEs also contain cell number parameters, denoted EC, NN, NA, M1 and M2, which are binary variables that represent cell activation. Monte Carlo steps are used to stochastically model interactions between agents, capturing the probabilistic nature of biological processes. The parameters and constants were all extracted from the code used to construct the agent-based model, and no modifications were made to the PDEs [17]. Fixed gradient boundary conditions (Neumann) are assumed indicating that the derivative at the boundary remains zero or constant.

2.2. Cytokine concentrations data

This dataset comprises ABM simulation results of cytokine concentrations captured over time, featuring columns denoting spatial coordinates (x,y) and concentrations of six cytokines: IL-8, IL-1 β , IL-6, IL-10, TNF- α , and TGF- β 1. Each row in the dataset corresponds to a specific time point, with 10^6 Monte Carlo steps (mcsteps) equivalent to exactly 100 h of simulated time. The equivalence of 10000 mcsteps to 1 h was an assumption of the authors [17]. The spatial coordinates represent the positions where cytokine concentrations are measured within the simulated biological environment. Cytokines are crucial signalling molecules for the regulation of immune response, inflammation, and cellular communication and movement. Consequently, each row of concentrations of cytokines within the dataset reflects the immune status or response of the system under study for a given time, at a given position. The columns of the dataset are shown in Table 1.

2.3. CellType positions data

CompuCell3D generates .vtk (Visualisation Toolkit) files that contain CellType positions in the defined grid. These files store data in a structured format, making them suitable for scientific computation and visualisation. They can handle different types of data, including scalar, vector, tensor, and field data.

In this research, we used .vtk files as they contain CellType information, which allows us to extract cell positions that are not directly available in our cytokine concentration data files.

Table 1
Cytokines dataset features and their values range.

Features	Range of values
mcsteps	[0 - 1000000] (increments of 10000)
x	[0 - 500]
y	[0 - 500]
il8	[0 - 7.75×10^{-9}]
il1	[0 - 3.83×10^{-9}]
il6	[0 - 4.48×10^{-9}]
il10	[0 - 1.24×10^{-9}]
tnf	[0 - 8.19×10^{-9}]
tgf	[0 - 6.38×10^{-9}]

2.4. Preparing the data for neural networks

The neural network models should return the six cytokine concentrations for the entire wound site per time step. Consequently, we need to formulate our input and output appropriately. Initially, we constructed 101 three-dimensional arrays, each corresponding to a timestep, with a shape of (50,50,6). Here, 50×50 represents the 2D spatial coordinates and 6 denotes the number of cytokine concentration features. Subsequently, we constructed a five-dimensional tensor with a shape of (99,2,50,50,6), where 99 indicates the number of timesteps, 2 represents the sequence length, and (50,50,6) corresponds to the spatial and feature dimensions of the arrays. This tensor structure was designed to serve as input for the neural networks, allowing the model to learn from sequences of two timesteps to predict cytokine concentrations for the next timestep. Additionally, we created a four-dimensional tensor with the shape of (99,50,50,6), which contains the target timesteps and their respective 3D arrays. This setup is essential for training the models to accurately predict the values of an array based on the preceding 2 arrays, thereby facilitating time-series forecasting of cytokine concentrations.

2.5. Data splits

It is crucial to ensure that the temporal or sequential nature of the data is preserved during the train-test split. A random split might break the sequential dependency and lead to data leakage between training and test sets. Instead of using a random split, we decided to use a time series-aware split. To handle this, we used a fixed training, validation, and testing split, producing a 70/10/20 split. We trained the NNs for two different splits, evaluating the data from $t = 82$ to $t = 100$ and from $t = 72$ to $t = 89$ (hours).

2.6. Long short-term memory neural networks

Long short-term memory (LSTM) neural networks, a specific type of recurrent neural network (RNN), are prevalent in literature for surrogate modelling tasks that involve capturing temporal dependencies. We implemented and compared the following four LSTM architectures:

- Long Short-Term Memory (LSTM) model [19];
- Long Short-Term Memory model incorporating convolutional layers (C-LSTM) [20];
- A more advanced C-LSTM architecture with time-distributed layers and max-pooling (CT-LSTM) [20];
- Spatio-Temporal Attention Long Short-Term Memory model incorporating attention mechanisms in the traditional LSTM architecture (STA-LSTM) [21]

The architecture has been optimised to better suit our problem (kernels, amount of layers, neurons per layer). The LSTM models are summarised in Table 2, and a simplified architecture is shown in Fig. 1.

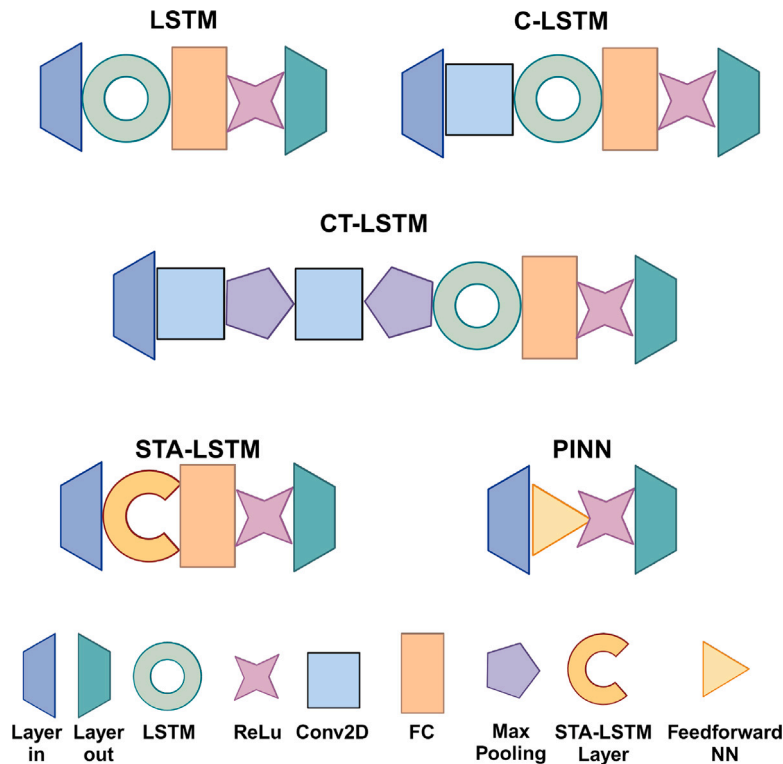


Fig. 1. Schematic representation of neural network architectures.

Table 2

Neural networks parameters and summary.

Models	LSTM	C-LSTM	CT-LSTM	STA-LSTM	PINN
Batch size	100	32	16	4	-
Output gate activation	ReLU	ReLU	ReLU	ReLU	ReLU
Normalisation	BatchNorm	BatchNorm	BatchNorm	BatchNorm	-
Epochs	1000	1000	1000	1000	10 000
Initial learning rate	1e-2	1e-3	1e-3	1e-6	1e-3
Loss function	MSE	MSE	MSE	MSE	Physics Loss
Regularization	L2 (0.03)	L2 (0.03)	L2 (0.01)	L2 (0.01)	-
Kernel Size	-	3 × 3	3 × 3	-	-
Max Pooling	-	-	2 × 2	-	-

2.7. Physics-informed neural network

Given their ability to incorporate physical laws into the learning process, it is possible that PINNs are able to outperform traditional RNN approaches in this domain.

Our PINN incorporates physical laws by solving the diffusion equations described in Section 2.1. MSE is measured separately and combined with the physics loss. The variables that represent cell activations, that were necessary to compute the physics loss, were not readily available from the tabular data containing cytokines concentrations, so we had to extract the position of different cells from the .vtk files to then create arrays that contain binary values that denote cell activations for every timestep.

2.8. Performance metrics

The performance of all neural networks was evaluated using statistical metrics: R-squared (R^2), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE). Additionally, we developed a custom accuracy function that triggers when the prediction is within 30 percent of the actual values. It is

important to note that MSE also served as the loss function for all neural networks, since it outperformed other loss functions (MAPE, MAE) in our experiments.

2.9. Implementation

Our neural networks were developed and trained using Python 3.8. TensorFlow 2.10 was used to construct and train the neural networks, as well as DeepXDE 1.10.1 [22] to construct the PINN. CUDA 11.2 and cuDNN 8.1 were used for parallel computation. We used the Dutch National Supercomputer Snellius for computing power to run different grid size simulations in CompuCell3D and machine learning experiments [23]. The neural networks were specifically trained on A100 GPUs.

3. Results

Tables A.3 and A.4 summarise the results for each model for the different data splits, where they are also more clearly shown in barplots (Fig. 2). The higher values are better for R^2 and accuracy, while the lower values are better for the error metrics.

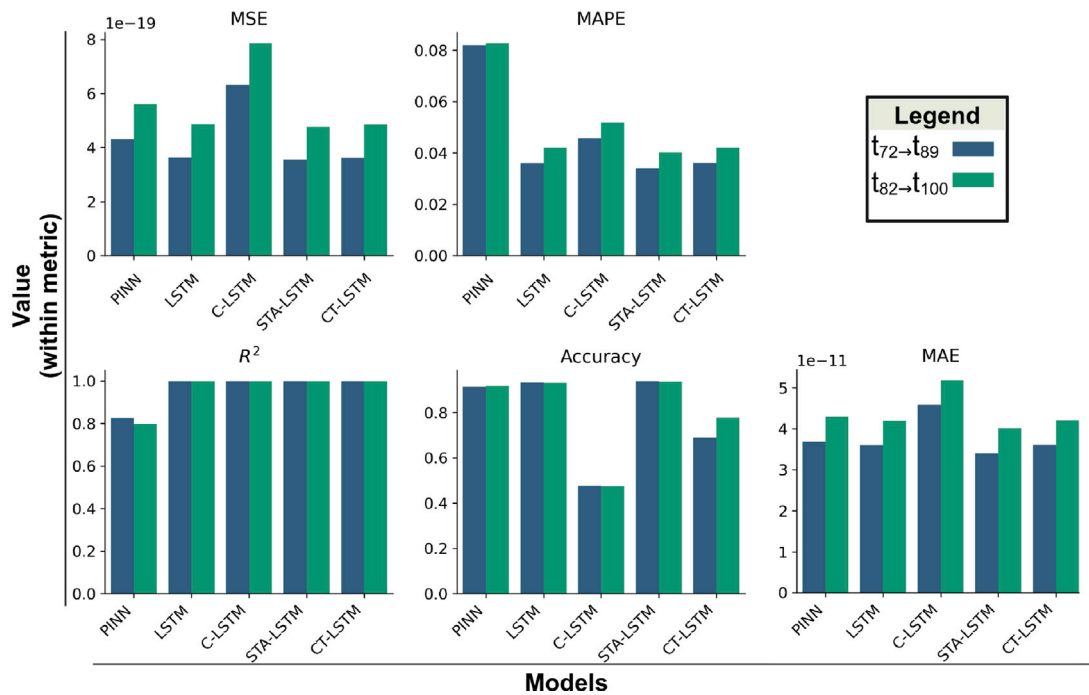


Fig. 2. Performance metrics for neural networks on test set (time $t = 72-89$ hr and $t = 82-100$ hr) for gridsize 250×250 . The metrics are also summarised numerically in Table A.3.

3.1. STA-LSTM outperforms other neural networks across most metrics

STA-LSTM generally performs the best in statistical metrics, especially excelling in R^2 , which measures how well the model explains the variance in the data, and MAPE, which reflects lower percentage errors. CT-LSTM, while performing well in our custom accuracy function, suffers from higher MAPE, suggesting some large errors. LSTM offers balanced performance with reasonable accuracy and error metrics. In contrast, C-LSTM consistently performs the worst in most metrics, indicating a worse fit and less accurate predictions. In general, STA-LSTM emerges as the best performing model in performance metrics.

3.2. C-LSTM significantly outperforms other neural networks at capturing spatial dependencies

We plotted the spatial concentration of predicted cytokines versus actual cytokine concentration at time $t = 72, 82, 92$ and 100 for all neural networks (Fig. 3). We also calculated the binary spatial accuracy of the predictions for each point in the grid in each time step (shown in Table A.5). The neural networks that most accurately capture the spatial distribution of cytokines in the grid are displayed in order (from top to bottom).

3.3. PINN predictions produce significantly larger standard deviation than the LSTMs

We computed the mean values and standard deviation per timestep per cytokine and then plotted the time series of the mean values for IL-8, IL-6, and TGF- β 1 (Fig. 4) across the last 19 timesteps of the simulation (from $t = 82$ to $t = 100$). This provides insight into how our neural networks capture the behaviour of the system across time for each cytokine. We choose to plot C-LSTM and STA-LSTM which are our worst and best-performing LSTM models in the statistical metrics as well as the PINN. The ranges of values for the concentrations of cytokines are plotted on the y -axis and the time steps that range from $t = 82$ to $t = 100$ are plotted on the x -axis.

3.4. Poor computational scalability for C-LSTM and CT-LSTM

We trained all neural networks for 50×50 , 100×100 , 250×250 , and 500×500 grid dimensions, where the performance metrics are summarised in Tables A.3, A.4. We also computed the average values across grid dimensions of our custom accuracy function for all 5 neural networks (Fig. 5), where we noticed C-LSTM and CT-LSTM fail at larger grid sizes due to their architecture being optimised during training locally at 50×50 grid dimensions.

4. Discussion

4.1. Training

It is important to use Rectified Linear Unit (ReLU) as an activation function, at the output gate, across all neural networks to guarantee non-negative predictions as concentrations cannot be negative. Using Exponential Linear Unit (ELU) or linear activation would seemingly capture outliers and a wider range of values better but also generate negative predictions. Other activation functions were also tested but produced much worse results. We split the data in such a way that respects the temporal nature of our dataset, but this kind of split makes it difficult for the models to generalise on unseen data in relation to some cytokines (such as TNF, TGF- β 1) as they appear later in the simulation. This issue can be addressed by making appropriate changes to the ABM code to run the ABM for two million mcsteps instead of 1 million. Leveraging transfer learning techniques by pre-training the models on earlier segments of the simulation and fine-tuning them for later stages may also help mitigate this generalisation gap and improve predictions for cytokines that manifest later in the process, but for the purposes of this research, this kind of split works well enough in assessing the neural networks.

4.2. Reflections on neural networks' performance

The way we formulate our input and output pairs for the sequential models is uncommon in the literature in other surrogating modelling

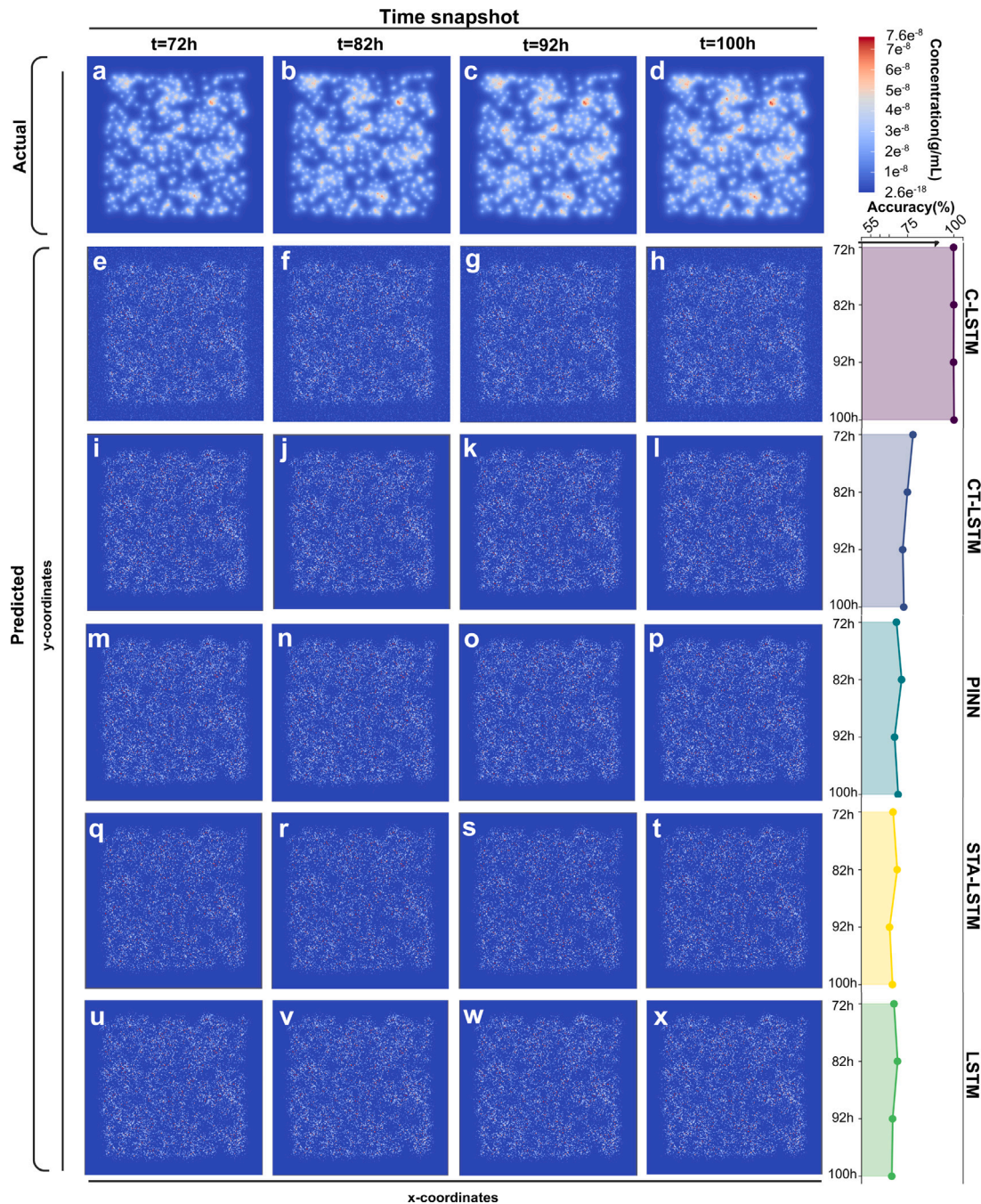


Fig. 3. Spatial plots for neural networks at time steps $t = 72, 82, 92,$ and 100 showing cytokine (for all 6) concentration of actual vs predicted values in order of accuracy (top to bottom). The computed binary spatial accuracy per method is displayed on the right. The results are also summarised in [Table A.5](#).

approaches [13–16], but uniquely suited to our problem due to the spatiotemporal nature of our data and the need to return values for the entire grid. Long short-term memory neural networks are widely established for time-series forecasting, especially in surrogate modelling contexts for ABMs [13,24]. C-LSTM models were originally conceptualised for text classification [20], but later used for a wide range of tasks such as detecting anomalies in web traffic [25] or detecting tumours [26]. As our C-LSTM model originally showed interesting results in some tasks, such as capturing spatial dependencies in the spatial plots (Fig. 3), we decided to construct a second neural network with a more complicated architecture and named it 'CT-LSTM' for the purposes of this paper. STA-LSTM models have been previously used in

tasks such as traffic forecasting [27] and flood forecasting [28,29], but there are no implementations for a related domain.

STA-LSTM seems to perform best in statistical metrics, but it is important to note that this is achieved by predicting IL-8 better than the other neural networks (Fig. 4), a cytokine that is at equilibrium during the evaluation of either data split. Other models, such as C-LSTM, make much stronger predictions for the more scarce cytokines that come later during simulation, such as IL-6 and TNF α .

In a biological context, PINNs have previously been used to solve problems such as Biot's equations [30], but there are not as many surrogate modelling approaches involving PINNs to solve diffusion equations. Our PINN, even with an 'out of the box' implementation,

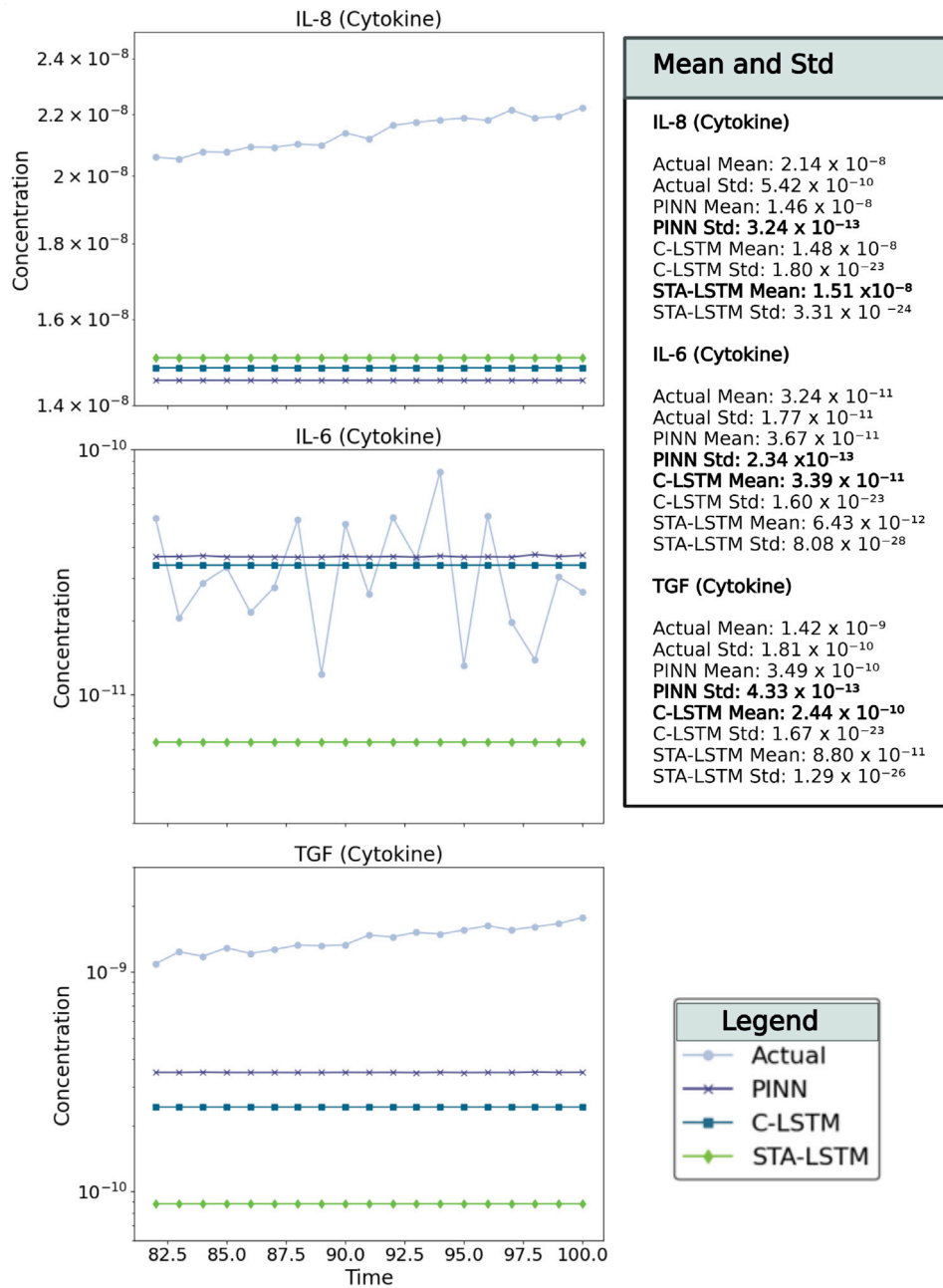


Fig. 4. Time-series of mean cytokine concentrations (IL-8, IL-6, and TGF) for C-LSTM, STA-LSTM, and PINN at 100×100 grid dimensions. Means and standard deviations for this time period are given in boxes. Those cytokines were specifically chosen to show the variability in IL-8 prediction, the systematic bias for TGF, and the poor prediction of STA-LSTM for IL-6 values. The best neural networks' predictions for Mean and Std are highlighted in bold.

seems to perform comparably to the best LSTM models, producing comparable metrics in MSE, MAE, and our custom accuracy function. However, it should be noted that MAPE is noticeably higher for the PINN compared to the LSTM networks. That is to be expected, as the PINN appears to be able to capture a more variable range of values of cytokine concentrations over time, as observed in the time series plots and also represented in the standard deviation (Fig. 4). It would also make sense for the PINN to produce a worse R^2 in larger grid spaces, as there is more data available then and the PINN focusses more on making sure that the loss adheres to physical laws, rather than fitting the data. Interestingly, the PINN also performs very well in the spatial plots outperforming neural networks that do not incorporate convolutional layers (Fig. 3). It is not surprising that neural networks

that incorporate convolutions would perform best in this task, as those architectures were originally conceptualised for text classification tasks.

4.3. Limitations

Our experimental setup was constructed to be able to train the neural networks locally but due to the small grid dimensions, we notice a number of problems. There is a sharp decrease in our custom accuracy metric in neural networks incorporating convolutional layers (Fig. 5), implying that the models predict more values that deviate significantly from the actual values. This makes sense as these models were optimised to capture a 50×50 grid space when performing convolutions.

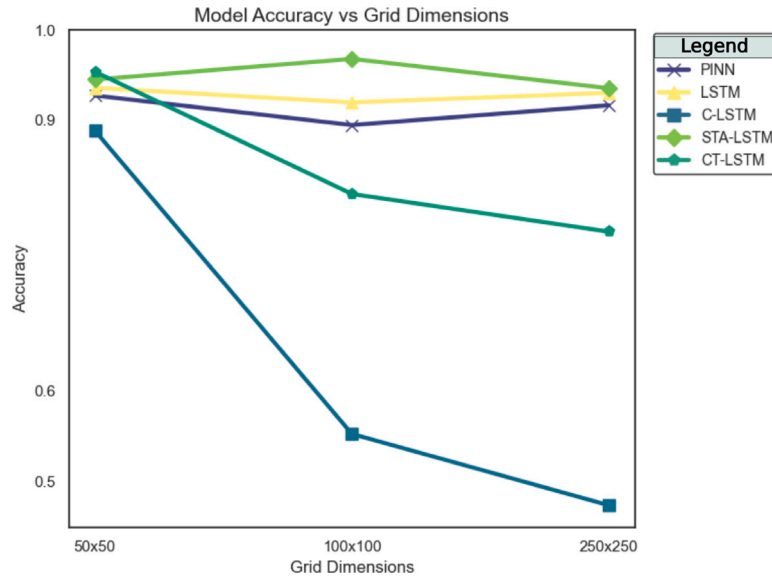


Fig. 5. Computed averages of neural networks' accuracy vs grid dimensions. C-LSTM and CT-LSTM produce significantly worse accuracy at higher grid dimensions due to their architecture being optimised locally at lower grid space.

Table A.3

Performance metrics for long short-term memory and physics-informed neural networks on test set ($t = 82$ to $t = 100$) for different grid dimensions. Missing data, due to model computational constraints is represented by “-”.

Metric	Model	50 × 50	100 × 100	250 × 250	500 × 500
R ²	PINN	0.8408	0.8683	0.7987	0.6646
	LSTM	0.8989	0.9950	0.9999	-
	C-LSTM	0.9007	0.9950	0.9999	-
	STA-LSTM	0.9697	0.9955	0.9999	0.9999
	CT-LSTM	0.9035	0.9951	0.9999	-
Accuracy	PINN	0.9276	0.8949	0.9168	0.9349
	LSTM	0.9361	0.9199	0.9306	-
	C-LSTM	0.8886	0.5528	0.4738	-
	STA-LSTM	0.9453	0.9682	0.9357	0.9243
	CT-LSTM	0.9539	0.8186	0.7768	-
MSE	PINN	1.1462×10^{-13}	4.9493×10^{-16}	5.6025×10^{-19}	4.6492×10^{-21}
	LSTM	1.0829×10^{-13}	4.5537×10^{-16}	4.8647×10^{-19}	-
	C-LSTM	1.0632×10^{-13}	4.5772×10^{-16}	7.8630×10^{-19}	-
	STA-LSTM	4.5794×10^{-14}	4.1275×10^{-16}	4.7718×10^{-19}	4.2133×10^{-21}
	CT-LSTM	1.0338×10^{-13}	4.4803×10^{-16}	4.8506×10^{-19}	-
MAPE	PINN	9.3963	1.8876	0.0828	0.0064
	LSTM	1.7844	0.5539	0.0420	-
	C-LSTM	2.2294	0.5563	0.0519	-
	STA-LSTM	1.4568	0.4336	0.0402	0.0027
	CT-LSTM	1.7538	0.5542	0.0421	-
MAE	PINN	2.9691×10^{-8}	1.6660×10^{-9}	4.2982×10^{-11}	2.6551×10^{-12}
	LSTM	2.9136×10^{-8}	1.6001×10^{-9}	4.1986×10^{-11}	-
	C-LSTM	2.9149×10^{-8}	1.5944×10^{-9}	5.1863×10^{-11}	-
	STA-LSTM	1.5172×10^{-8}	1.3998×10^{-9}	4.0182×10^{-11}	2.7315×10^{-12}
	CT-LSTM	2.8463×10^{-8}	1.5908×10^{-9}	4.2104×10^{-11}	-
Physics Loss	PINN	8.20×10^{-14}	3.72×10^{-16}	3.96×10^{-19}	3.39×10^{-21}

This is an issue of scalability for our models, where models incorporating convolutional layers would require additional optimisation at higher grid dimensions. To address this issue, hierarchical frameworks combining CNNs with LSTM layers present a promising direction as they have been used for similar tasks such as short-term travel speed prediction [31], demonstrating the ability to handle spatiotemporal dependencies across multiple scales. In another domain, stock market forecasting models also leverage hierarchical ConvLSTM structures to predict continuous trends by analysing sequential data with dynamic resolution [32].

The range of our predictions is limited from $t = 10$ to $t = 100$, where the assumption is made that there would be no practical need to predict values so early in the simulation, as it would be easy to run

the simulation up to that point. Furthermore, with the current train-test split, the data is not trained for all timesteps, due to the sequential nature of our setup, which causes systematic bias in predicting values for some cytokines such as TGF- β 1 (Fig. 4).

5. Conclusions and future work

Our current methods for comparing NN architectures can be expanded to other RNNs such as Gated Recurrent Units (GRUs) [33], or more complicated architectures that incorporate convolutions and recurrent networks, such as the ConvLSTM architecture [34], or ConvGRUs [35]. It is also possible to expand the current implementation of PINN with recently published work [36], where PINNs are combined with DeepONets [37].

Table A.4

Performance metrics for long short-term memory and physics-informed neural networks on test set ($t = 72$ to $t = 89$) for different grid dimensions. Missing data, due to model computational constraints is represented by “-”.

Metric	Model	50 × 50	100 × 100	250 × 250	500 × 500
R ²	PINN	0.8597	0.8657	0.8264	0.7101
	LSTM	0.9152	0.9962	0.9999	-
	C-LSTM	0.9211	0.9961	0.9999	-
	STA-LSTM	0.9475	0.9968	0.9999	0.9999
	CT-LSTM	0.9232	0.9963	0.9999	-
Accuracy	PINN	0.9306	0.8987	0.9137	0.9471
	LSTM	0.9389	0.9217	0.9325	-
	C-LSTM	0.8828	0.5182	0.4748	-
	STA-LSTM	0.9384	0.9708	0.9389	0.9411
	CT-LSTM	0.9715	0.8230	0.6879	-
MSE	PINN	9.1306×10^{-14}	4.6434×10^{-16}	4.3120×10^{-19}	3.4794×10^{-21}
	LSTM	8.6572×10^{-14}	3.6769×10^{-16}	3.6367×10^{-19}	-
	C-LSTM	8.0579×10^{-14}	3.7251×10^{-16}	6.3030×10^{-19}	-
	STA-LSTM	6.8073×10^{-14}	3.0353×10^{-16}	3.5400×10^{-19}	3.2894×10^{-21}
	CT-LSTM	7.8398×10^{-14}	3.5867×10^{-16}	3.6197×10^{-19}	-
MAPE	PINN	9.3812	1.7791	0.0820	0.0063
	LSTM	1.6467	0.4711	0.0360	-
	C-LSTM	2.0896	0.4787	0.0459	-
	STA-LSTM	1.5418	0.3407	0.0340	0.0024
	CT-LSTM	1.5682	0.4698	0.0361	-
MAE	PINN	2.6277×10^{-8}	1.4724×10^{-9}	3.6925×10^{-11}	2.2657×10^{-12}
	LSTM	2.5779×10^{-8}	1.3660×10^{-9}	3.5990×10^{-11}	-
	C-LSTM	2.5164×10^{-8}	1.3657×10^{-9}	4.5908×10^{-11}	-
	STA-LSTM	2.2943×10^{-8}	1.1059×10^{-9}	3.3976×10^{-11}	2.3619×10^{-12}
	CT-LSTM	2.4493×10^{-8}	1.3501×10^{-9}	3.6081×10^{-11}	-
Physics Loss	PINN	1.26×10^{-13}	4.95×10^{-16}	6.13×10^{-19}	4.95×10^{-21}

Table A.5

Computed binary spatial accuracy for neural networks at different timesteps for 250 × 250 grid dimensions.

Model	Timesteps					
	72	76	82	92	96	100
C-LSTM	99.62%	99.23%	99.75%	99.65%	99.64%	99.89%
CT-LSTM	77.82%	78.89%	74.75%	72.24%	73.16%	72.85%
PINN	68.92%	72.72%	71.69%	67.88%	69.83%	69.83%
LSTM	67.54%	70.66%	69.49%	66.82%	68.53%	66.37%
STA-LSTM	67.17%	69.63%	69.36%	65.29%	68.41%	66.82%

The insights extracted from the comparison of neural networks in this paper can be used to construct a more thorough implementation of this system that can ultimately provide clinicians with a better understanding of cytokine dynamics that can inform therapeutic strategies. Clinicians, specifically burn surgeons, rely on key cytokine levels, cell counts, and other biomarkers to make informed decisions on the treatment strategy and care requirements. Model predictions on such cytokine levels at the wound level, which are seldom available experimentally, would provide the clinicians with a window-view into the burn characteristics. This enhanced data availability would therefore provide additional clarity towards the treatment strategy decisions and consequent prevention of post-burn complications.

Neural networks will naturally have a degree of error in their predictions when implemented in regression problems. To deal with uncertainty in predictions, it is possible to construct multiple PINNs to then ensemble predictors. There is also value in studying a comparison of aggregation functions between models, as that is also seemingly an open research question in the context of surrogate modelling for ABMs.

This paper demonstrated the potential of integrating neural networks (NNs) as surrogate models to enhance the computational efficiency of an agent-based model (ABM) of the immune response to burn wounds. Using the strengths of both computational modelling techniques and neural networks, our objective was to address the computational limitations inherent in ABMs while maintaining prediction accuracy. We implemented and assessed four different architectures of the LSTM model and constructed an architecture for a physics-informed neural network (PINN), where each model was evaluated

using a comprehensive set of performance metrics. Our findings indicate that different NN architectures can simulate ABM results with varying degrees of accuracy and computational speed. Although all models tested massively improved computational efficiency, the extent of this improvement and the trade-off with prediction accuracy varied depending on the task, where C-LSTM achieved significantly better results in capturing spatial dependencies, while PINN produced a standard deviation that better reflected the expected variability in individual predictions, and STA-LSTM achieved better performance in statistical metrics.

CRedit authorship contribution statement

Ioannis Papapanagiotou: Writing – review & editing, Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Roland V. Bumbuc:** Writing – review & editing, Visualization, Validation, Supervision, Software, Formal analysis, Data curation, Conceptualization. **H. Ibrahim Korkmaz:** Writing – review & editing, Validation, Resources, Project administration, Funding acquisition, Conceptualization. **Valeria Krzhizhanovskaya:** Writing – review & editing, Visualization, Validation, Conceptualization. **Vivek M. Sheraton:** Writing – review & editing, Validation, Supervision, Software, Resources, Project administration, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is a follow-up study from a Master Thesis entitled “Assessing performance of long short-term memory and physics-informed neural networks for surrogate modelling of burn wounds immune response”.

Funding

The authors declare that financial support was received for the research, authorship, and/or publication of this article. This research was partly funded by the Dutch Burns Foundation (DBF) – Health-Holland, Beverwijk, Project PPS 22.01.

Appendix A. Tables

See Tables A.3–A.5.

Data availability

The code and data required to reproduce our experiments are registered on Zenodo[38].

References

- [1] L.H. Evers, D. Bhavsar, P. Mailänder, The biology of burn injury, *Exp. Dermatol.* 19 (9) (2010) 777–783, <http://dx.doi.org/10.1111/j.1600-0625.2010.01105.x>.
- [2] R. Gallinaro, W. Cheadle, K. Applegate, H. Polk, The role of the complement system in trauma and infection, *Surg. Gynecol. ; Obs.* 174 (5) (1992) 435–440, URL <http://europepmc.org/abstract/MED/1570624>.
- [3] A. Presbitero, E. Mancini, R. Brands, V.V. Krzhizhanovskaya, P.M.A. Sloot, Supplemented alkaline phosphatase supports the immune response in patients undergoing cardiac surgery: Clinical and computational evidence, *Front. Immunol.* 9 (2018) <http://dx.doi.org/10.3389/fimmu.2018.02342>.
- [4] R.N. Germain, M. Meier-Schellersheim, A. Nita-Lazar, I.D. Fraser, Systems biology in immunology: A computational modeling perspective, *Annu. Rev. Immunol.* 29 (1) (2011) 527–585, <http://dx.doi.org/10.1146/annurev-immunol-030409-101317>, PMID: 21219182.
- [5] H.I. Korkmaz, F.B. Niessen, A. Pijpe, V.M. Sheraton, F.J. Vermolen, P.A. Krijnen, H.W. Niessen, P.M. Sloot, E. Middelkoop, S. Gibbs, et al., Scar formation from the perspective of complexity science: a new look at the biological system as a whole, *J. Wound Care* 31 (2) (2022) 178–184, <http://dx.doi.org/10.12968/jowc.2022.31.2.178>.
- [6] A. Presbitero, V. Melnikov, V. Krzhizhanovskaya, P. Sloot, A unifying model to estimate the effect of heat stress in the human innate immunity during physical activities, *Sci. Rep.* 11 (2021) 16688, <http://dx.doi.org/10.1038/s41598-021-96191-0>.
- [7] R.V. Bumbuc, V. Yildirim, M.V. Sheraton, Modelling the interplay between chronic stress and type 2 diabetes on-set, in: *International Conference on Computational Science*, Springer, 2023, pp. 330–338, http://dx.doi.org/10.1007/978-3-031-36021-3_34.
- [8] P. Kaura, T. Mishra, N. Verma, I.S. Dalal, V. Sheraton, Effects of combined chemotherapeutic drugs on the growth and survival of cancerous tumours–an in-silico study, *J. Comput. Sci.* 54 (2021) 101421, <http://dx.doi.org/10.1016/j.jocs.2021.101421>.
- [9] R.V. Bumbuc, H.I. Korkmaz, P.P.M. van Zuijlen, L. Vermeulen, V.M. Sheraton, Understanding the dynamics of the proliferative phase in local burn wound healing: A computational model, in: *2023 IEEE International Conference on Bioinformatics and Biomedicine, BIBM*, 2023, pp. 3676–3683, <http://dx.doi.org/10.1109/BIBM58861.2023.10385875>.
- [10] J.J. DiStefano, Dynamic systems biology modeling and simulation, in: *Dynamic Systems Biology Modeling and Simulation*, 2015, pp. 100–200, URL <https://api.semanticscholar.org/CorpusID:267787367>.
- [11] G. Ermentrout, L. Edelstein-Keshet, Cellular automata approaches to biological modeling, *J. Theoret. Biol.* 160 (1) (1993) 97–133, <http://dx.doi.org/10.1006/jtbi.1993.1007>.
- [12] F. Chiacchio, M. Pennisi, G. Russo, S. Motta, F. Pappalardo, Agent-based modeling of the immune system: NetLogo, a promising framework, *BioMed Res. Int.* 2014 (2014) 907171, <http://dx.doi.org/10.1155/2014/907171>.
- [13] D. Larie, G. An, R.C. Cockrell, The use of artificial neural networks to forecast the behavior of agent-based models of pathophysiology: An example utilizing an agent-based model of sepsis, *Front. Physiol.* 12 (2021) <http://dx.doi.org/10.3389/fphys.2021.716434>.
- [14] J.T. Nardini, Forecasting and predicting stochastic agent-based model data with biologically-informed neural networks, *Bull. Math. Biol.* 86 (11) (2024) 130, <http://dx.doi.org/10.1007/s11538-024-01357-2>.
- [15] C. Angione, E. Silverman, E. Yaneske, Using machine learning as a surrogate model for agent-based simulations, *PLOS ONE* 17 (2) (2022) 1–24, <http://dx.doi.org/10.1371/journal.pone.0263150>.
- [16] S. van der Hoog, Surrogate modelling in (and of) agent-based models: A prospectus, *Comput. Econ.* 53 (2018) <http://dx.doi.org/10.1007/s10614-018-9802-0>.
- [17] H.I. Korkmaz, V.M. Sheraton, R.V. Bumbuc, M. Li, A. Pijpe, P.P.G. Mulder, B.K.H.L. Boekema, E. de Jong, S.G.F. Papendorp, R. Brands, E. Middelkoop, P.M.A. Sloot, P.P.M. van Zuijlen, An in silico modeling approach to understanding the dynamics of the post-burn immune response, *Front. Immunol.* 15 (2024) <http://dx.doi.org/10.3389/fimmu.2024.1303776>.
- [18] M.H. Swat, G.L. Thomas, J.M. Belmonte, A. Shirinifard, D. Hmeljak, J.A. Glazier, Multi-scale modeling of tissues using CompuCell3D, in: *Methods in Cell Biology*, vol. 110, Elsevier, 2012, pp. 325–366.
- [19] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.* 9 (8) (1997) 1735–1780, <http://dx.doi.org/10.1162/neco.1997.9.8.1735>.
- [20] C. Zhou, C. Sun, Z. Liu, F.C.M. Lau, A C-LSTM neural network for text classification, 2015, [arXiv:1511.08630](https://arxiv.org/abs/1511.08630).
- [21] S. Song, C. Lan, J. Xing, W. Zeng, J. Liu, An end-to-end spatio-temporal attention model for human action recognition from skeleton data, in: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI '17*, AAAI Press, 2017, pp. 4263–4270, URL <https://dl.acm.org/doi/10.5555/3298023.3298186>.
- [22] L. Lu, X. Meng, Z. Mao, G.E. Karniadakis, DeepXDE: A deep learning library for solving differential equations, *SIAM Rev.* 63 (1) (2021) 208–228, <http://dx.doi.org/10.1137/19M1274067>.
- [23] H. Bal, D. Epema, C. de Laat, R. Van Nieuwpoort, J. Romein, F. Seinstra, C. Snoek, H. Wijshoff, A medium-scale distributed system for computer science research: Infrastructure for the long term, *Computer* 49 (5) (2016) 54–63, <http://dx.doi.org/10.1109/MC.2016.127>.
- [24] S. Wang, K. Fan, N. Luo, Y. Cao, F. Wu, C. Zhang, K. Heller, L. You, Massive computational acceleration by using neural networks to emulate mechanism-based biological models, *Nat. Commun.* 10 (2019) 1–9, <http://dx.doi.org/10.1038/s41467-019-12342-y>.
- [25] T.-Y. Kim, S.-B. Cho, Web traffic anomaly detection using C-LSTM neural networks, *Expert Syst. Appl.* 106 (2018) 66–76, <http://dx.doi.org/10.1016/j.eswa.2018.04.004>.
- [26] Y. Liu, Y.-X. Huang, X. Zhang, W. Qi, J. Guo, Y. Hu, L. Zhang, H. Su, Deep C-LSTM neural network for epileptic seizure and tumor detection using high-dimension EEG signals, *IEEE Access* 8 (2020) 37495–37504, <http://dx.doi.org/10.1109/ACCESS.2020.2976156>.
- [27] R. Jiang, H. Xu, G. Gong, Y. Kuang, Z. Liu, Spatial-temporal attentive LSTM for vehicle-trajectory prediction, *ISPRS Int. J. Geo- Inf.* 11 (7) (2022) <http://dx.doi.org/10.3390/ijgi11070354>.
- [28] Y. Ding, Y. Zhu, J. Feng, P. Zhang, Z. Cheng, Interpretable spatio-temporal attention LSTM model for flood forecasting, *Neurocomputing* 403 (2020) 348–359, <http://dx.doi.org/10.1016/j.neucom.2020.04.110>.
- [29] Y. Ding, Y. Zhu, Y. Wu, F. Jun, Z. Cheng, Spatio-temporal attention LSTM model for flood forecasting, in: *2019 International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2019, pp. 458–465, <http://dx.doi.org/10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00095>.
- [30] T. Kadeethum, T.M. Jørgensen, H.M. Nick, Physics-informed neural networks for solving nonlinear diffusivity and Biot’s equations, in: F.-B. Tian (Ed.), *PLOS ONE* 15 (5) (2020) e0232683, <http://dx.doi.org/10.1371/journal.pone.0232683>.
- [31] W. Wang, X. Li, Travel speed prediction with a hierarchical convolutional neural network and long short-term memory model framework, 2018, [arXiv:1809.01887](https://arxiv.org/abs/1809.01887).
- [32] A. Chakraborty, A. Basu, A hierarchical conv-LSTM and LLM integrated model for holistic stock forecasting, 2024, [arXiv:2410.12807](https://arxiv.org/abs/2410.12807).
- [33] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014, [arXiv:1412.3555](https://arxiv.org/abs/1412.3555).
- [34] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, W.-c. Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in: *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS '15*, MIT Press, Cambridge, MA, USA, 2015, pp. 802–810, URL <https://dl.acm.org/doi/10.5555/2969239.2969329>.
- [35] X. Shi, Z. Gao, L. Lausen, H. Wang, D.-Y. Yeung, W.-k. Wong, W.-c. WOO, Deep learning for precipitation nowcasting: A benchmark and a new model, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), in: *Advances in Neural Information Processing Systems*, vol. 30, no. 1, Curran Associates, Inc., 2017, URL https://proceedings.neurips.cc/paper_files/paper/2017/file/a6db4ed04f1621a119799fd3d7545d3d-Paper.pdf.
- [36] B. Lin, Z. Mao, Z. Wang, G.E. Karniadakis, Operator learning enhanced physics-informed neural networks for solving partial differential equations characterized by sharp solutions, 2023, [arXiv:2310.19590](https://arxiv.org/abs/2310.19590).
- [37] L. Lu, P. Jin, G. Pang, Z. Zhang, G.E. Karniadakis, Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, *Nat. Mach. Intell.* 3 (3) (2021) 218–229, <http://dx.doi.org/10.1038/s42256-021-00302-5>.
- [38] I. Papapanagiotou, R.V. Bumbuc, H.I. Korkmaz, V. Krzhizhanovskaya, V.M. Sheraton, giannis3p/neural-agent-models: From simulations to surrogates: Neural networks enhancing burn wound healing predictions, 2024, <http://dx.doi.org/10.5281/zenodo.14003270>.