Scientific Eminence Group

Journal of Biodegradation and Bioremediation

Research Article

Open Access

jbb@scientificeminencegroup.com

Nested Machine Learning for Modelling Degradation and Drug Release of Bioresorbable Polymers

Mingxuan Xia[1]*, Ran He[1], Peter Polak[1], Shuihua Wang[2], Venkat Ghantasala[1], Kevser Sevim[3] and Baber Saleem[1] and Jingzhe Pan[1]

[1]*School of Engineering, University of Leicester, Leicester, LE1 7RH, United Kingdom*

[2]*Department of Biological Sciences, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, 215123, China*

[3]*Quotient Sciences, Nottingham, NG11 6JS, United Kingdom*

*Corresponding Author

Mingxuan Xia, School of Engineering, University of Leicester, Leicester, LE1 7RH, United Kingdom, E-mail: mx54@leicester.ac.uk

Citation

Mingxuan Xia, Ran He, Peter Polak, Shuihua Wang, Venkat Ghantasala (2025) Nested Machine Learning for Modelling Degradation and Drug Release of Bioresorbable Polymers. J Biodegrad BioRem 3: 1-18

Publication Dates

Received date: December 27, 2024

Accepted date: January 27, 2025

Published date: January 30, 2025

Abstract

This paper presents a novel approach of computer modelling to predict degradation of and drug release from bioresorbable polymers. Various uncertain elements in a mathematical model for polymer degradation and drug release are replaced by artificial neural networks (ANNs). Two examples of the uncertain elements are (a) the dependence of diffusion coefficients on concentration and porosity and (b) the order of reaction rate for hydrolysis leading to polymer chain cleavage. The ANNs are trained, while being nested in the mathematical model, to minimise the error of predictions for molecular weight, mass loss and drug release. This effectively provides the mathematical model, in the format of partial differential equations, with the ability of machine learning. The approach was firstly applied to a well-known case in the literature on the size effect of degradation of poly-lactide-co-glycolide (PLA/GA). Secondly the interaction between a basic drug and the hydrolysis chain session of PLA/GA was considered. It was demonstrated that the model with nested ANNs can learn from the experimental data to make accurate predictions.

Keywords: Polymer Degradation; Drug Release; Modelling; Artificial Neural Network; Nested Machine Learning

Introduction

Following some early setbacks in developing bioresorbable coronary stents [1], major efforts are being made to learn the lesion and develop the next generation of bioresorbable implants. Some recent examples can be found in refs [2-4]. Polyesters such as polylactides (PLA) and their copolymers with polyglycolides (PGA) are the most commonly used biodegradable polymers for medical implants due to their susceptibility to hydrolytic degradation within the human body. The degradation products of these polyesters are generally non-toxic and can be safely absorbed or excreted by the body, minimising long-term biocompatibility issues. This process begins with the diffusion of water molecules into the amorphous regions of the polymer, initiating the cleavage of ester bonds. As hydrolysis progresses, the molecular weight of the polymer reduces and short chains diffuse out of the polymer leading to mass loss of the implant. Eventually the polymers are metabolised into carbon dioxide and water through pathways in the Krebs cycle. The degradation rate of the polyesters can be effectively controlled by adjusting material factors such as initial molecular weight and crystallinity, enabling them to meet various biomedical needs, such as drug delivery and tissue engineering. The degradation rate is also significantly influenced by local and environmental factors, including the pH of bodily fluids, enzymatic activity and local tissue inflammation. In addition, embedded drugs in the implants can affect the degradation of the polymers - the hydrogen ions from acidic drugs or the hydroxide ions from basic drugs can alter the hydrolysis rate significantly. Due to the long-term nature of the degradation (from a few months to several years), tailoring the degradation and drug release rates through trial-and-error experiments is a major challenge in developing clinical applications [3,4].

Mathematical models have been developed to describe the degradation of polyesters [5-7] and drug release from these polymers [8,9]. A comprehensive treatment of these mathematical models can be found in the book edited by Pan [10]. A recent example of applications of these models in stent design can be found in [11]. Typically, these models are in the format of a set of partial differential equations (PDEs) which can be solved using commercial finite element software for practical design cases [12]. It has been demonstrated that these models can capture the general trends of device degradation and drug release, but often fail to precisely fit with the experimental data. The inaccuracy is caused by various uncertainties in the mathematical models. A typical example is the diffusion coefficient (of short oligomers or drug molecules) which depends on the concentration of the diffusing species and the porosity of the diffusion medium, both of which changes significantly as the polymer degrades. Accurately calibrating the dependence directly from experiment is challenging and often impractical. Another example is the nature of the rate equations for the underlying chemical reactions. The first order reaction is often assumed which may not reflect the real chemistry of the underpinning process.

A potential alternative approach of modelling is to use machine learning. A very active area of research is to replace a set of partial differential equations (PDEs) with artificial neural networks (ANNs). Different types of ANNs have been designed and trained using numerical solutions of PDEs leading to the so-called physics-informed neural networks (PINNs). As a relevant example to this work, Li *et al.* [13] used a convolutional neural network to learn the finite element solutions of a reaction-diffusion equation. All these works took the PDEs as the "ground truth", which is not necessarily the truth at all because of the uncertain elements, such as the diffusion coefficient mentioned above, used in the PDEs.

The central idea of this paper is to replace the various uncertain elements in the mathematical model by artificial neural networks (ANNs). We propose to combine the certain elements of the model, such as matter conservation and chemical equilibrium, with ANNs that replace the uncertain elements. The benefit of nesting ANNs in the model is that they can be trained using experimental data to eliminate the uncertainties. There are two challenges in taking this approach. The first one is that training an ANN requires a huge amount of data while data for long term polymer degradation are very expensive and time consuming to obtain. The second challenge is that the existing training algorithms are not designed for ANNs that are nested inside mathematical equations. Using the diffusion coefficient again as an example, training an ANN to learn its dependence on porosity would require experimental data of the diffusion coefficient at various levels of porosity. In reality only mass loss and molecular weight are measured in long-term degradation experiments.

This paper presents a novel method for training the ANNs nested in a set of mathematical equations to overcome the

two challenges. The method was applied firstly to a well--known case in the literature on the size effect of degradation of PLA/GA [14] and then secondly to a case of drug release from PLA/GA microspheres in the literature [15]. In order to model the drug release case, the interaction between a basic drug and the hydrolysis chain session of PLA/GA was introduced into the mathematical model. The two case studies clearly demonstrated the self-learning capacity of our proposed approach by using only limited experimental data.

A Model for Interaction Between Basic Drug and Polyester Degradation

Sevim and Pan considered the case of acidic drugs releasing from microspheres of PLGA [8]. Here we consider the case of basics drugs release. The governing equations developed by Pan and co-workers [7,8,10,11] for degradation of bioresorbable polyester are modified to consider the interaction between the polyesters and basic drugs. The hydrolysis reaction can be phenomenologically described as

$$Polyster + H_2O \overset{H+}{\Leftrightarrow} R - COOH + R - OH, \#(1)$$

which states that ester bonds are broken down into carboxylic acids (R-COOH) and alcohols (R-OH). The carboxylic end groups have a high degree of acid disassociation which can be described as

$$\mathrm{R\text{ - }COOH} \overset{\mathrm{R\text{ - }CO}}{\Leftrightarrow} \mathrm{O^- + H^+}, \#(2)$$

The dissociation of basic drug molecules can be phenomenologically expressed as

$$\mathrm{Drug + H_2O \Leftrightarrow Drug_{H^+} + OH^-}, \#(3)$$

The disassociation of the carboxylic acid end groups and that of the basic drug molecules are much faster than the hydrolysis reaction of the polyesters. Consequently, they are both assumed to be in equilibrium.

Assuming water is abundant (they diffuse into an implant much faster than the implant degradation), the chain scission rate of the polyester due to the hydrolysis reaction can be empirically written as:

$$\frac{\partial R_s}{\partial t} = k_1 C_e + k_2 C_e C_{\mathrm{H^+}} + k_3 C_e C_{\mathrm{OH^-}}, \#(4)$$

where R_s is the molar number of chain scissions per unit volume, C_e, C_{H+} and C_{OH-} represent the molar concentrations of ester bonds, hydrogen ions and hydroxide ions respectively, k_1, k_2 and k_3 are the phenomenological rate constants. The first term on the right-hand side represents non-catalyzed hydrolysis, the second term represents acid-catalyzed hydrolysis, while the last term represents the interaction between drug and polymer.

The hydrogen ions (H^+) can come from three sources: (a) the surrounding solution in the order of 10^{-pH}, (b) reversible hydrolysis of the drug molecules in the order of $K_bC_{drug,m}/C_{OH-}$, where k_b is the acid dissociation constant for the drug molecules and $C_{drug,m}$ is the mole concentration drug molecules, and (c) the acid disassociation of the carboxylic acid ends, which can be calculated as

$$C_{H^+} = \frac{K_a C_{R-COOH}}{C_{R-COO^-}}, \#(5)$$

where K_a is the acid dissociation constant. Simple calculations reveal that (c) dominates C_{H+} by at least three orders of magnitude, hence Eqn. (5) is used to calculate C_{H+} in Eqn. (4).

COH- in Eqn. (4) can be calculated from the equilibrium expression as

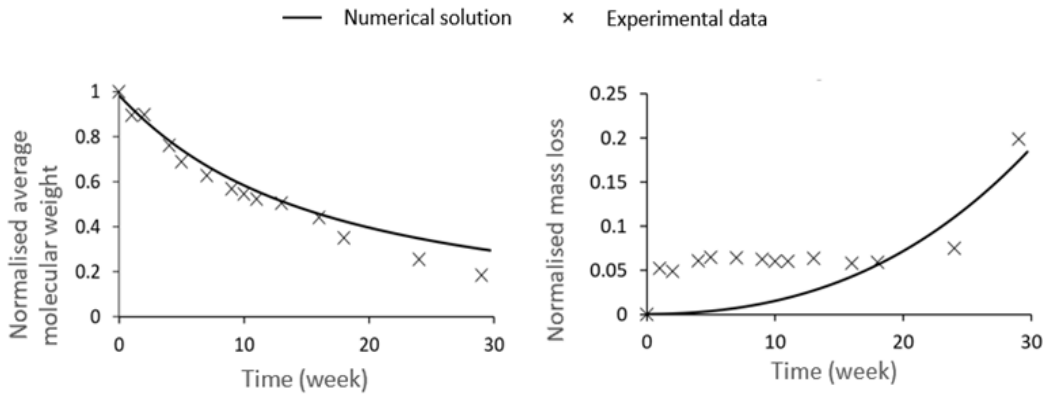
$$C_{\mathrm{OH}^-} = (K_b \times C_{drug,m})^{0.5}, \#(6)$$

The rest of the governing equations for computing (a) the molecular weight of the polyester, (b) the mass loss due to short chain diffusion and (c) drug diffusion follow the previous work by Pan and co-workers [7,8,10,11]. For readers convenience, the full set of equations are provided in Appendix A.

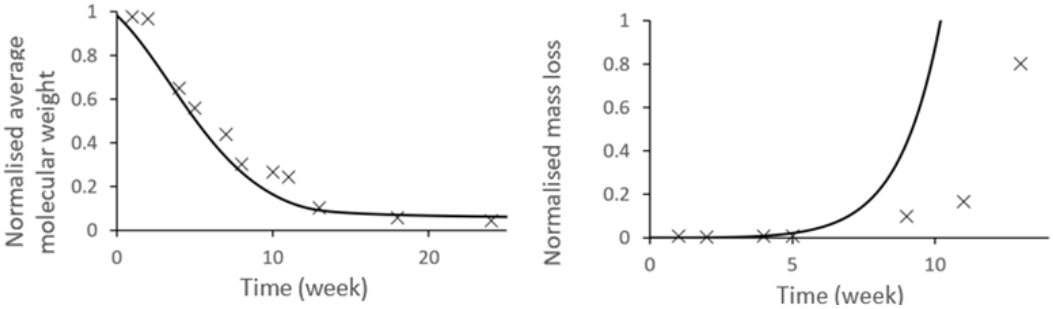
The Need for Machine Learning

In a classical case study, Grizzi *et. al.* [14] demonstared that thicker plates of PLA/GA degrade faster than thinner ones due to the autocatalytic effect. The acid disassociation of the polymer chain ends increases the local acidity and leads to faster degradation. In the thin plates, short chains are able to diffuse out which reduces the autocatalytic effect. Grizzi *et. al.* were the first to highlight the size effect of PLA/GA degradation. In a systematic study, Fitzgerald and Corrigan [15] mesaured both the degradtion of and the levamisole release from their PLA/GA mcrospheres. They demonstrated that the basic drug accelerates the polymer degradation. These two complementary work are used in this paper as our case studies.

Figure 1 and Figure 2 present our best effort in fitting the experimental data by using the mathematical model presented in Appendix A. Five cases are presented here including the degradation of PLA/GA plates of thicknesses of 0.3 mm (case A) and 2 mm [14] (case B), and the degradation of PLA/GA microspheres and drug release of loadings of levamisole at 2.4% (case C), 14.3% (case D), and 19.7% (case E) [15]. For the plates, the equations were solved for the one-dimensional reaction-diffusion problem while for the micro-spheres, spherical symmetry was assumed in the numerical solution. A central finite difference scheme was used to discretize the second terms on the right-hand side of Eqns. (A-6) and (A-7). An ODE solver in MATLAB was used for the time integration. The kinetic parameters in the model (see Table B in Appendix B) were varied manually over a very large range to search for those that give the best fit. The further details of the numerical solution and the set of model parameters that generated the numerical solutions shown in Figs.1 and 2 are provided in Appendix B.

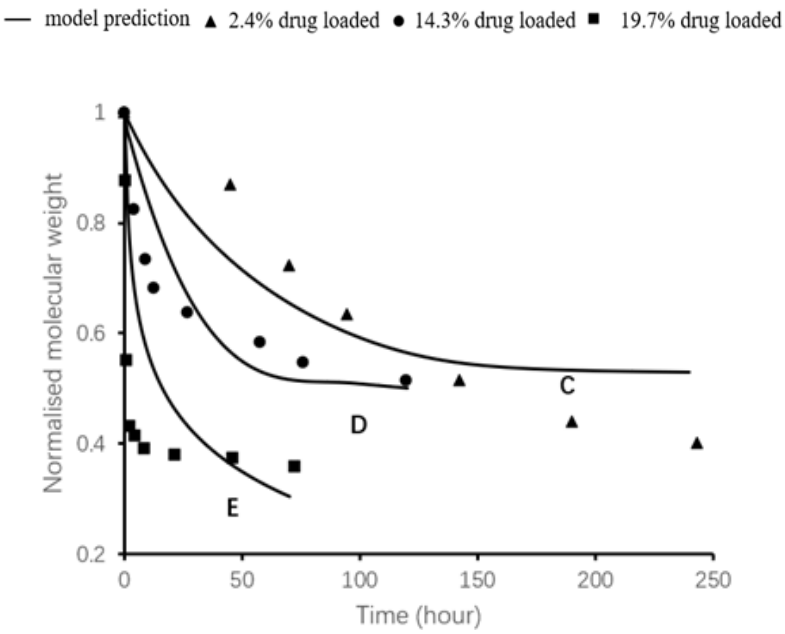


(a) Case A – PLA/GA plates of 0.3mm in thickness

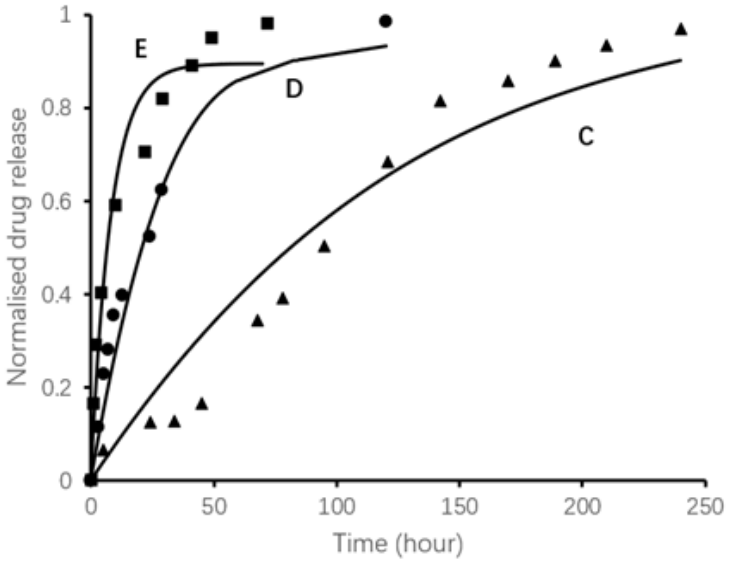


(b) Case B – PLA/GA plates of 2mm in thickness

Figure **1:** Temporal evolutions of normalised average molecular weight and normalised mass loss for cases A and B, numerically calculated by solving Eqns. (A.1)-(A.10), in comparison with experimental data in [14]. Solid lines represent the model prediction and discrete symbols represent corresponding experimental data.



(a) Molecular weights as function of time



(b) Levamisole release as function of time

Figure 2: Temporal evolutions of normalised average molecular weight (Fig 2.a) and normalised drug release (Fig 2.b) for drug loadings of levamisole at 2.4% (case C), 14.3% (case D), and 19.7% (case E), which were numerically calculated by solving Eqns. (A.1)-(A.10), in comparison with experimental data in [15]. Solid lines represent the model prediction and discrete symbols represent corresponding experimental data.

It can be observed from these figures that the mathematical model is only good enough as an approximate guide in most the cases. The governing equations in Appendix A consist of two parts: (I) principles that cannot be violated such as matter conservation during chemical reaction and diffusion, Eqns (A.6) and (A.7), and (II) empirical rules such as the reaction rate equation (A.1), the expressions for diffusion coefficient, Eqns (A.9), (A.30), and the rate equation for oligomer production of Eqn. (A.3). The validity of these empirical rules depends on many factors and therefore represents a high degree of uncertainty in the mathematical model. In this paper it is proposed that these empirical rules are replaced by artificial

neural networks which are summarised in Table 1.

Table 1: Input and output of feed forward neural networks (FNNs)

Sub-model	inputs	output	Number of layers
Chain scission rate (Eqn. (A.1))	$C_e,\ C_{ol},\ C_{drug}$	$\frac{dR_s}{dt}$	5
Oligomer production rate (Eqn. (A.3))	$C_e, R_s, \frac{dR_s}{dt}$	$\frac{dR_{ol}}{dt}$	4
Drug diffusion coefficient (Eqn. (A.9))	$D_{drug,poly}, D_{drug,pore}, V_{pore}$	D_{drug}	5
Oligomer diffusion coefficient (Eqn. (A.8))	$D_{poly}, D_{pore}, V_{pore}$	D_{ol}	5

Figure 3 illustrate how a feedforward neural network was used in our model. In forward predictions, the ANN processes input data through multiple layers to generate a final output. Firstly, the input data is fed into the network, where each feature is passed to neurons in the hidden layers. Each neuron processes the input by applying a weighted sum with biases to it and then passing the result through an activation function, such as ReLU or Sigmoid, to introduce non-linearity. The output from each neuron is then sent to the next layer, repeating the process through all hidden layers produces a prediction. Effectively the ANN is just a set of mathematical operations that calculate an output variable, e.g. the chain session rate, from the input concentrations of the reaction species. Once the weights and biases are known, it is straightforward to implement the ANN in the mathematical model. The challenge is to obtain the weights and biases that lead to accurate predictions of the mathematical model, a process known as training or back-propagation in machine learning.

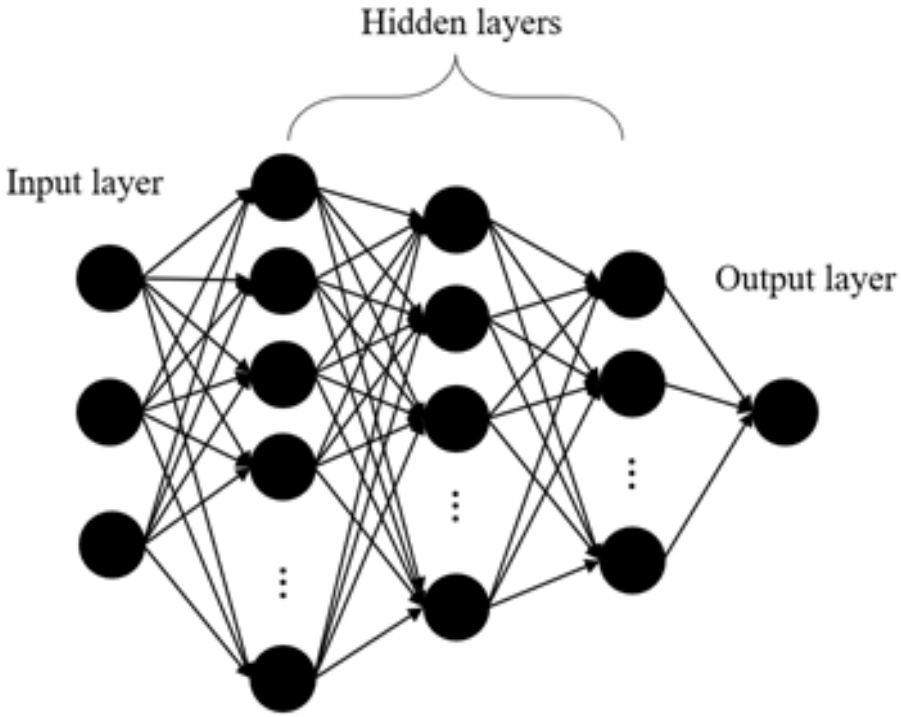


Figure 3: Schematic representation of *FNN* architecture

Nested Machine Learning

Training an ANN requires a huge amount of data. Fortunately the ANNs have the ability of transfer learning, that is, they can accumulate knowledge by updating the weights and biases repeatedly. An ANN can be trained firstly using data generated from analytical theories instead of experimental data, and then re-trained using limited experimental data. We have demonstrated this two-step training approach in a previous work [16]. This approach is adopted here, the ANNs shown in **Table 1** were all firstly trained by data generated using the corresponding analytical equations (presented in Appendix A). Although these empirical equations may not be fully accurate, they offer a solid foundation by capturing the initial relationship between variables in the model.

In the second step, the pre-trained ANNs are implemented into the governing equations. Unfortunately, all the existing back-propagation algorithms were designed for minimising a loss function that is defined by the outputs of the ANN. In our case the loss function can only be defined by using the output of the entire mathematical model instead that of the ANN. For example, experimental data exists only for drug release rather than the drug diffusion coefficient. Consequently, the loss function can only be defined by the difference between the predicted and measured drug release profiles. A training algorithm is required for the ANN nested in the model to update its weights and biases to minimising this difference.

Backpropagation Algorithm for Training ANNs Nested in a Finite Element Package

An ANN nested in the governing equations can be trained by minimising the mean squared error of

$$J((\hat{M}-M)) = \frac{1}{2N}\sum_{(a=1)}^{A}(\hat{M}^a - M^a)^2, \# \quad (7)$$

Where **M** is the experimental data for the measured variable, $\hat{M}$ is the vector of the corresponding variable estimated by the governing equations with a nested ANN, *A* is the total number of experimental data and ***N*** is the batch size. Specifically, $\hat{M}$ can be calculated from the governing equations with the nested ANN. In this paper, **M** = [*normalised molecular weight, normalised mass loss, normalised drug release*].

The training dataset for the ANN is denoted as $\left\{\left(x^1, y^1\right), \ldots, \left(x^n, y^n\right)\right\}$. Using the chain session rate as an example, $x = scaled\ [C_e, C_{ol}, C_{drug}]\ and y = scaled\ [\frac{dR_s}{dt}]$. For each $\left(x^k, y^k\right)\ (1 \leqslant k \leqslant n)$, the components of the vectors in the hidden and output layers can be calculated by

$$z_j^{lk} = f\left(z0_j^{lk}\right) = f\left(\sum_i \left(w_{ji}^l z_i^{(l-1)k}\right) + b_j^l\right), \quad (8)$$

where l indicates the layer, f (z) is the activation function and $w_{ji}^l and\ b_j^l$ are the components of the weights and biases in the l layer, respectively. Hence, $z_i^{1k} = x_i^k\ and\ z_i^{ok} = \hat{y}_i^k, where\ x_i^k\ and\ \hat{y}_i^k$ are the components of the input and ANN's estimated output vectors, respectively.

Each iteration of gradient descent updates the weights according to

$$w_{ij_{new}}^l = w_{ij_{old}}^l - \alpha J_{ij}^{\prime l}, \quad (9)$$

where α is the learning rate and $J_{ij}^{\prime l}$ is the partial derivative of *J* in terms of the weights in the *l* layer. Considering the output layer, we have

$$J'^{o}_{ij} = \frac{\partial J}{\partial w^{o}_{ij}} = \sum_{a=1}^{A} \left\{ \frac{\partial J}{\partial \left(\hat{M}^{a} - M^{a} \right)} \sum_{k} \frac{\partial \hat{M}^{a}}{\partial \hat{y}^{k}_{i}} \frac{\partial \hat{y}^{k}_{i}}{\partial w^{o}_{ij}} \right\}. \quad (10)$$

Calculating $\partial \hat{M}^{a} / \partial \hat{y}^{k}_{i}$ is a major challenge due to the intricate nature of the governing equations. However, upon careful examination of Eqn. (10), it becomes apparent that *J'* will multiply the learning rate α, which is empirical in machine learning. Certain algorithms, such as Adam, adjust the learning rate dynamically throughout the training process. Consequently, the precise value of $\partial \hat{M}^{a} / \partial \hat{y}^{k}_{i}$ becomes irrelevant, whilst its sign plays a critical role. It is therefore proposed that one doesn't calculate the actual value of $\partial \hat{M}^{a} / \partial \hat{y}^{k}_{i}$ and use α to incorporate its effect. Consequently eqn. (10) is rewritten as

$$J'^{o}_{ij} = \pm \sum_{a=1}^{A} \left\{ \frac{\partial J}{\partial \left(\hat{M}^{a} - M^{a} \right)} \sum_{k} \frac{\partial \hat{y}^{k}_{i}}{\partial w^{o}_{ij}} \right\}, \quad (11)$$

where the sign of ± is determined by the relationship between the variables measured in the experiment (such as drug release) and the output variables of the ANN (such as drug diffusion coefficient). Since these variables have physical meanings, for each specific chemical or physical process under consideration, the sign can be determined by logic derivation and then checked by numerical calculations. This leads to a very simple backpropagation algorithm of Eqns. (9) and (11) for ANNs nested in the governing equations.

The terms in Eqn. (11) can be calculated by

$$\frac{\partial J}{\partial \left(\hat{M}^{a} - M^{a} \right)} = \delta^{oa} = \frac{1}{N} \left(\hat{M}^{a} - M^{a} \right), \quad (12)$$

$$\frac{\partial \hat{y}^{k}_{i}}{\partial w^{o}_{ij}} = \frac{\partial f\left(z0^{ok}_{j} \right)}{\partial z0^{ok}_{j}} z^{(o-1)k}_{j}. \quad (13)$$

Similarly, the partial derivatives of the hidden layers can be written as

$$J'^{l}_{ij} = \pm \sum_{a=1}^{A} \left\{ \frac{\partial J}{\partial \left(\hat{M}^{a} - M^{a} \right)} \sum_{k} \sum_{p=1}^{r^{o}} \frac{\partial \hat{y}^{k}_{p}}{\partial w^{l}_{ij}} \right\} = \pm \sum_{a=1}^{A} \left\{ \frac{\partial J}{\partial \left(\hat{M}^{a} - M^{a} \right)} \sum_{k} \sum_{p=1}^{r^{o}} \frac{\partial \hat{y}^{k}_{p}}{\partial z^{lk}_{i}} \frac{\partial z^{lk}_{i}}{\partial w^{l}_{ij}} \right\}, \quad (14)$$

Where

$$\delta_i^{la} = \frac{\partial J}{\partial\left(\hat{M}^a - M^a\right)} \sum_{p=1}^{r^o} \frac{\partial \hat{y}_p^k}{\partial z_i^{lk}} = \sum_{b=1}^{r^{l+1}} \left\{ \left[\frac{\partial J}{\partial\left(\hat{M}^a - M^a\right)} \sum_{p=1}^{r^0} \frac{\partial \hat{y}_p^k}{\partial z_b^{(l+1)k}} \right] \frac{\partial z_b^{(l+1)k}}{\partial z_i^{lk}} \right\}$$

$$= \sum_{b=1}^{r^{l+1}} \delta_b^{(l+1)a} \frac{\partial z_b^{(l+1)k}}{\partial z_i^{lk}}, \#(15)$$

with

$$\frac{\partial z_b^{(l+1)k}}{\partial z_i^{lk}} = \frac{\partial f\left(z0_i^{(l+1)k}\right)}{\partial z0_i^{(l+1)k}} w_{bi}^{(l+1)}, \quad (16)$$

$$\frac{\partial z_i^{lk}}{\partial w_{ij}^l} = \frac{\partial f\left(z0_i^{lk}\right)}{\partial z0_i^{lk}} z_j^{(l-1)k}, \quad (17)$$

where r^l is the number of vector components in the l layer.

The biases are updated using similar approach such that

$$b_{ij_{new}}^l = b_{ij_{old}}^l - \alpha \sum_{a=1}^{A} \sum_k \delta_i^{la} \frac{\partial z_i^{lk}}{\partial b_{ij}^l}, \quad (18)$$

$$\frac{\partial z_i^{lk}}{\partial b_{ij}^l} = \frac{\partial f\left(z0_i^{lk}\right)}{\partial z0_i^{lk}}. \quad (19)$$

The algorithm proposed above is generally valid for training ANNs nested in any partial differential equations (PDEs).

It is obvious that an increase in the chain session rate, $\frac{dR_s}{dt}$, leads to a decrease in molecular weight, M_n; an increase in oligomer production rate $\frac{dR_{ol}}{dt}$, leads to an increase in mass loss, and an increase in drug and short chain diffusion coefficients, Dol and D_{drug}, leads to an increase in drug release. Hence the corresponding signs in Eqn. (11) was – for chain session rate and positive for all the others.

Hyperparameters, Training Data and Refined Algorithm for the ANNs

The first and last two ANNs in **Table 1** consist of an input layer, three hidden layers with 64, 32 and 16 neurons respectively, and an output layer. The second ANN in **Table 1** comprises an input layer, two hidden layers with 32 and 16 neurons, and an output layer. These ANNs were implemented using the Keras-TensorFlow machine learning library. The training data were scaled using MinMaxScaler, AbsMaxScaler and QuantileTransformer from Python's sklearn library. Leaky ReLU activation functions were used in all hidden layers, and a linear activation function was used in the output layer.

For the pre-training, Adam algorithm was used to minimise the mean squared error of the outputs, with a batch size of 128 and 1000 epochs. The training dataset were generated using the analytical equations. The range of the dataset were determined from the numerical solutions presented in **Figure** 1 and **Figure 2**. Within this range, 30,000 sets of synthetic input data were randomly generated in Excel using the RANDARRAY function. Subsequently, the corresponding output values were computed using Eqns (A.1), (A.3), (A.8) and (A.9). This set of synthetic and labeled input-output data was utilized to train the ANNs shown in **Table 1**. Four separate datasets were created to accommodate the specific requirements of each ANN. All the datasets were split into training, validation, and test sets in an 80:10:10 ratio.

For the nested training, numerical solutions (forward prediction) for the molecular weight, mass loss and drug release as functions of time can be obtained by using a combination of the central finite difference scheme and MATLAB ODE solver. The loss function of Eqn. (7) can then be calculated at the discrete times when experimental data are available. Eqns. (9) and (11) can then be used to update the weights while Eqns. (18) can be used to update the biases. Due to the very limited number of experimental data points, a further refinement of the backpropagation scheme was necessary which consisted of the following steps:

- between any two experimental data points, values of the input variables for the ANNs (see Table 1) were generated from the forward prediction and recorded in a dataset for all the finite difference nodes and for all the time integration steps.
- between the two experimental data points, the value of the loss function is labeled against the first data point only.
- using this single value of the loss function, the weights and biases of the ANNs are repeatedly updated for the entire dataset of step (i).
- then the forward prediction is repeated using the updated weights and biases and the scheme moves to the next experimental data point.
- (i) to (iv) is repeated until the last experimental data point is reached.
- if the value of the loss function is still unacceptable, the entire process of (i) to (v) is repeated.

It was found that repeating the entire process (step vi) for over 1000 times is necessary for the nested training to be successful. The success of training an ANN can be easily judged by the fact that the obtained weights and biases can give correct predictions for the experimental data.

For cases A and B of PLA/GA plates, the ANNs replacing rate equations for chain session and oligomer production were retrained using the algorithm. The loss function is to minimise the difference between the predicted and measured values of the normalised average molecular weight and mass loss. For Cases C to E of drug release from the microspheres, all the ANNs presented in Table 1 were retrained using the algorithm. The loss function was extended to also minimizing the difference between the predicted and measured values of normalised drug release.

Results

Pre-training of ANNs Using Data Generated From Analytical Equations

To demonstrate the capacity of the ANNs in capturing the analytical equations, scatter plots of predicted values by the ANNs vs. true values (calculated from the equations, all normalised) are depicted for the rate of chain scission in **Figure 4**(a) and rate of oligomer production in **Figure 4**(b) respectively. 24000 data points were used in each of the figures. A data point falling on the line of 45 degrees means the ANN is precise at that pair of values. It can be observed that all the

points form a narrow black band well within 0.5% of the error band (shown by the two read lines).

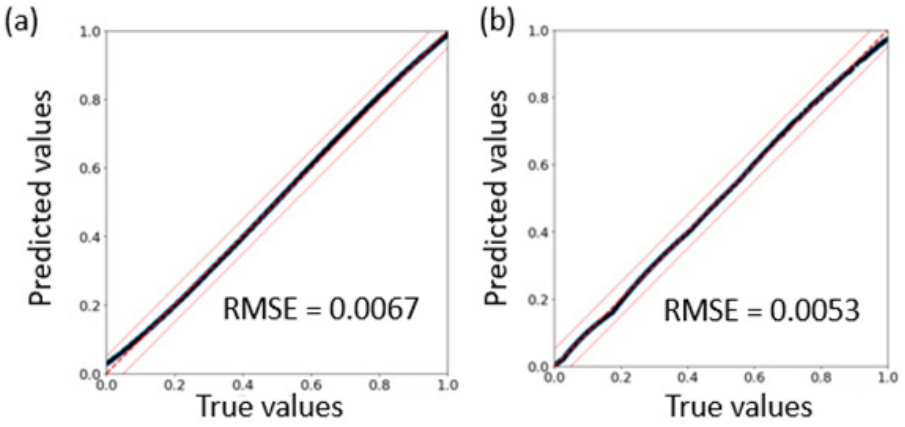


Figure 4: Scatter plots of predicted values vs. true values (normalised) for ANNs trained using data generated from the analytical equations for Case A (PLA/GA plates of 0.3mm in thickness); (a) rate of molar number of chain scission per unit volume and (b) rate of molar number of oligomer concentration per unit volume. The root mean squared errors (RSMEs) are also shown.

To demonstrate the feasibility of nesting ANNs in the mathematical model, Eqns. (A.1) and (A.3) were replaced by the trained ANNs and the equations were solved numerically. **Figure 5** compares the average molecular weight and mass loss obtained by solving the original equations (dash lines) and equations with nested ANNs (solid lines). It can be observed that the two set of results are very close to each other.

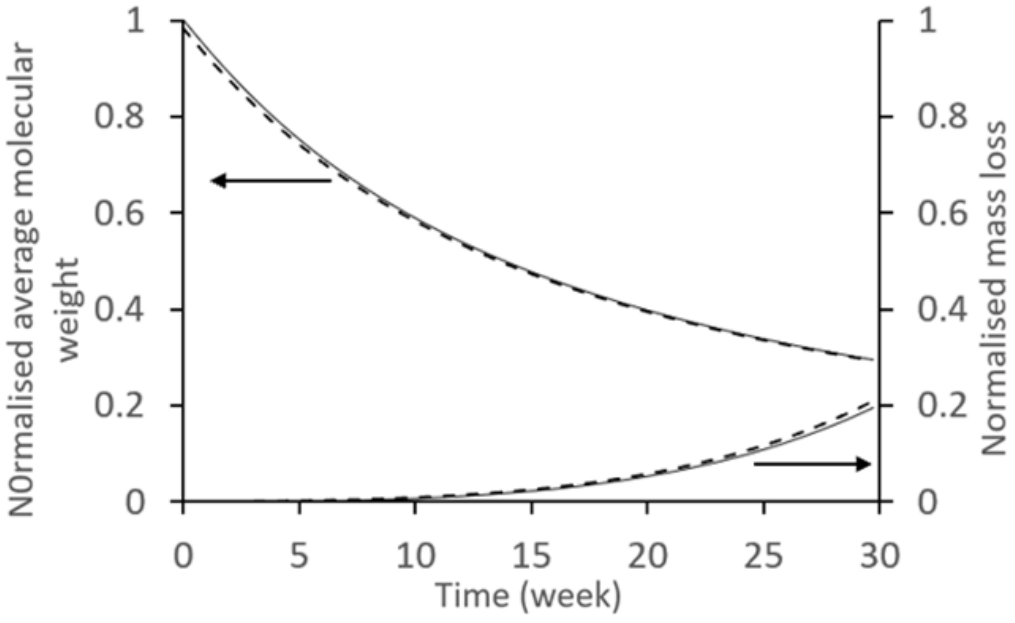
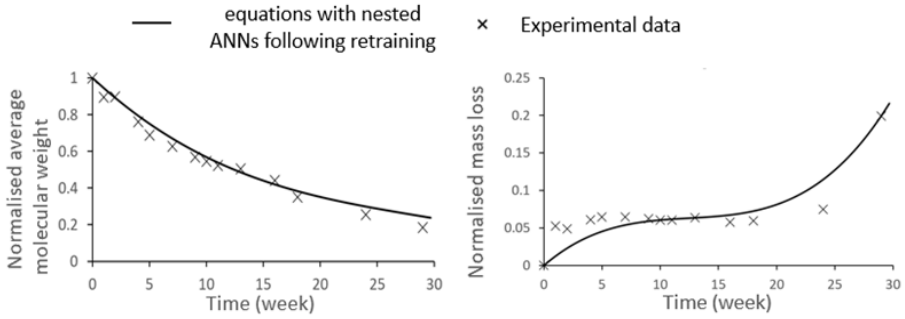


Figure 5: Temporal evolutions of normalised average molecular weight and mass loss for Case A obtained by solving the original equations (dash lines) and equations with nested ANNs (solid lines).

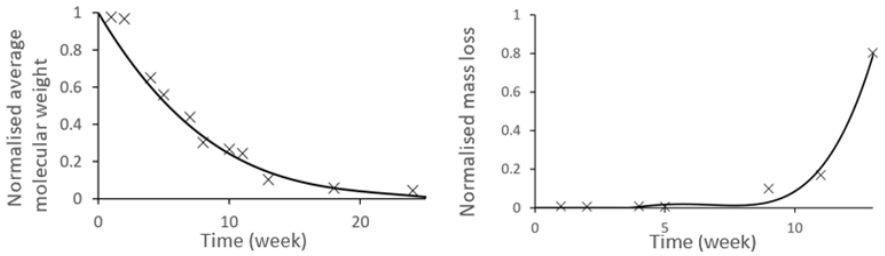
Nested Training of ANNs for Case A of PLA/GA Plates of Different Thicknesses

Figure 6 shows the temporal evolutions of average molecular weight and normalised mass loss of the PLA/GA plates of two different thicknesses predicted by the governing equations with ANNs nested, following their retraining by using the algorithm of section 4.2.The predictions are compared to the ex-

perimental data obtained from Ref [14]. It can be observed from the figures that the predictions have been significantly improved by comparing to those shown in **Figure** 1. In case A, the experimental data for mass loss at the very beginning of the degradation test was very strange. The sample lost 5% of its mass almost instantaneously which cannot be caused by polymer degradation (as degradation hasn't really started). It would be reasonable in the computer model for polymer degradation to ignore the initial mass loss (i.e. subtract all the data points by 5%). In this work the initial loss was deliberately kept to test the ability of the nested machine learning approach to learn from strange experimental data. It can be observed from **Figure 6**(b) that the model did try its best to provide a reasonable fit. If the first data point was ignored, we could indeed achieve a perfect fit for mass loss by retaining the ANN, similar to Case B.



(a) Case A – PLA/GA plates of 0.3mm in thickness



(b) Case B – PLA/GA plates of 2mm in thickness

Figure 6: Temporal evolutions of normalised average molecular weight and normalised mass loss for Cases A and B, predicted by nested ANNs following retraining in comparison with the experimental data [14]. Solid lines represent the model prediction and discrete symbols represent corresponding experimental data.

Table 2 provides a quantitative measure for the prediction accuracies by the model using nested ANNs in comparison with the experimental data. The average prediction error is defined as

$$APE = \frac{1}{d}\sum_{i=1}^{d}\left|\frac{\text{Predicted}_i - \text{Experiment}_i}{\text{Experiment}_i}\right|, \quad (20)$$

Where d is the number of experimental data points.

Table 2: Average prediction errors for Cases A and B (Figure 5)

		Case A	Case B
average molecular weight	Original equations	0.0517	0.1206
	With retrained ANNs	0.0420	0.1103
mass loss	Original equations	0.4224	5.5546
	With retrained ANNs	0.1330	0.0404

Nested Training of ANNs for Case C-E of Drug Loaded Microspheres

Figure 7 shows the temporal evolutions of average molecular weight (**Figure 7** (a)) and drug release (**Figure 7** (b)) for the three different drug loadings using model nested with the ANNs following their retraining. By comparing **Figure 7** with **Figure 2** (which used the original model), it can be observed that retraining the ANNs has significantly improved the mode accuracy. **Table 3** presents the average prediction errors for Cases C-E.

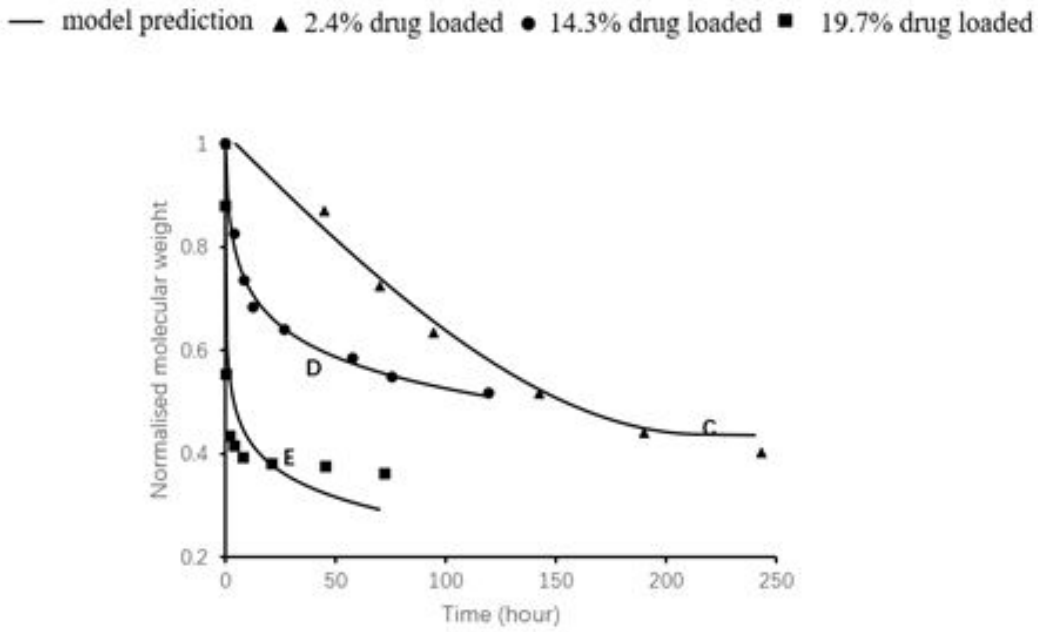


Figure 7: Temporal evolutions of normalised average molecular weight and normalised drug release for Cases C, D and E, predicted by nested ANNs following retraining in comparison with experimental data [15]. Solid lines represent the model prediction and discrete symbols represent corresponding experimental data

Table 3: Average prediction errors for Case C-E (**Figure 7**)

		Case C	Case D	CaseE
Average molecular weight	Original equations	0.2841	0.2679	0.3477
	With retrained ANNs	0.1007	0.1709	0.2318
Drug release	Original equations	0.0577	0.0906	0.3894
	With retrained ANNs	0.0328	0.0132	0.1445

Discussion

This study represents the first attempt to empower the mathematical models for polymer degradation with the ability of machine learning in order to eliminate their uncertainties. Such uncertainties are a norm instead of an exception when modelling polymer degradation and drug release in the complex biological environment inside a human body. The ability of machine learning for the mathematical models is a step change in the game. The Absorb Bioresorbable Vascular Scaffold (Abbott, USA) made of bioresorbable polymers was an exciting development in intervention cardiology but had to be withdrawn from the market in 2017 due to long-term complications after implementation [17,18]. Nevertheless, a large amount of extremely valuable data from long term animal and clinical trials have been accumulated. These data can be used to train the mathematical models with nested ANNs, leading to a powerful design tool for the next generation of bioresorbable implants.

This study validated the feasibility and capabilities of the proposed approach by accurately fitting the polymer degradation behaviour in the four different cases. The ANNs also demonstrated their capacity in the prediction of size effect in the polymer degradation in Case A and B, indicating their applicability across various geometries. In Case C-E, the impact of alkaline drug hydrolysis on polymer degradation was considered, which was not addressed in previous studies. By incorporating this factor, the model provided a more comprehensive understanding of the degradation process, which is crucial for optimising the design of bioresorbable implants. This level of accuracy was achieved autonomously by the ANNs, requiring minimal computational resources and time, in contrast to the time-consuming process of traditionally manual improvement of the empirical mathematical models.

A notable challenge in training an ANN is the demand for a huge amount of training data. To address this challenge, this study adopted a two-step training strategy, firstly training the ANNs to the learn the empirical equations, followed by re-training using experimental data. This strategy was also used in [16,19]. This study expanded the two-step learning strategy to a general situation where it was not practical to obtain experimental data for the direct output variables of the ANNs. In the case of degradation studies of bioresorbable medical implants, it was impractical to measure experimentally the change in diffusion coefficient or the polymer chain breakage and the consequent oligomer production. Long-term experimental data only exists for the average molecular weight, mass loss and drug release as functions of time. The results of our work showed that the two-step learning strategy using our backpropagation algorithm for ANNs nested in the model provides a effective solution for this problem of lack of direct experimental data.

In our study, attempt was also made to replace the Fick's First Law with an ANN. However, this did not lead to any significant improvement in the prediction accuracy, revealing that the Fick's First Law was valid in this particular application. This is not necessarily true for many other applications, and ANNs can always be developed and trained using the proposed methodology to replace the Fick's First Law where necessary.

It is important to note that the proposed approach and backpropagation algorithm are generally valid for all partial differential equations (PDEs). This opens the door for extending the model to include biological processes, such as tissue regeneration and differentiation, and interactions between tissue, released drugs and the degrading polymer, which would have far more uncertainty and therefore benefit even more from the nested machine learning.

Conclusions

This paper presents a new modelling approach for polymer degradation and drug release by nesting artificial networks (ANNs) in the mathematical equations. The nested ANNs re-

place empirical rules in the model, thereby combining the certainty of science with the ability of machine learning, leading to the removal of uncertainties in the empirical rules. A back propagation algorithm was presented for training ANNs nested in the governing equations. The case studies demonstrated that our mathematical model nested with ANNs can indeed learn to accurately predict the molecular weight change, mass loss and drug release during PLA/GA degradation. The approach proposed in this study is generic and valid for all mathematical models. It opens the door for confidently modelling interactions between drugs, tissues and implants in a complex biological environment.

Disclosure

The authors do not have any conflicts of interest to disclose.

Acknowledgments

This work was supported by the UKRI's Strength in Place Fund [grant number 82148]; University of Leicester; China Scholarship Council.

Appendix A. Governing equations for polyester degradation and drug release

As mentioned in Sec. 2, this study considered solely the effect of polymer hydrolysis on the hydrogen ion concentration. Hence, Eqn. (4) can be rewritten as

$$\frac{\partial R_s}{\partial t} = C_e(k_1 + k_2^{'}C_{\mathrm{H}^+} + k_3^{'}C_{\mathrm{OH}^-}), \#\,(A.1)$$

Where *Cdrug* is the volume concentration of the drug and C_e can be calculated by

$$C_e = C_{e0} - R_{ol}, \#\,(A.2)$$

Where R_{ol} is the molar concentration of all the produced oligomers given by

$$\frac{dR_{ol}}{dt} = \alpha\beta\left(\frac{R_s}{C_{e0}}\right)^{\beta-1}\frac{dR_s}{\mathrm{dt}}, \#\,(A.3)$$

and α and β are empirical constants. In Eqn. (4) and Eqn. (A-1), we have $k_2^{'}$ and $k_3^{'}$ are defined by

$$k_2^{'} = k_2^{\left(\frac{K_a}{m}\right)^{0.5}}, \#\,(A.4)$$

$$k_3^{'} = k_3^{\left(K_b\frac{C_{drug,m}}{C_{drug}}\right)^{0.5}}, \#\,(A.5)$$

and m is average degree of polymerisation of oligomers.

In the diffusion part of degradation process, short chains are randomly produced by the hydrolysis reaction of long chains. Those oligomers are diffused out by the concentration difference as the driven force. It is the same for diffusion of drug molecules. We have

$$\frac{\partial C_{drug}}{\partial t} = \frac{\partial R_{drug}}{\partial t} + \nabla \bullet (D_{drug}\nabla C_{drug}), \#\,(A.6)$$

$$\frac{\partial C_{ol}}{\partial t} = \frac{\partial R_{ol}}{\partial t} + \nabla \bullet (Dol\nabla C_{ol}), \#\,(\mathrm{A}.7)$$

where Eqn. (A-6) is for drug and Eqn. (A-7) is for oligomers, C_{ol} is the molar number of ester bonds in oligomers per unit volume, C_{drug} is the mole concentration of drug and R_{drug} is the mole concentration of drug dissolution

In which diffusion coefficients of oligomers and drug molecules are defined as

$$D_{ol=}D_{poly} + \left(1.3{V_{pore}}^2 - 0.3{V_{pore}}^3\right)\left(D_{pore} - D_{poly}\right), \#\,(A.8)$$

$$D_{drug} = D_{drug,poly} + \left(1.3{V_{pore}}^2 - 0.3{V_{pore}}^3\right)\left(D_{drug,pore} - D_{drug,poly}\right), \#\,(A.9)$$

Where V_{pore} is porosity and calculated by

$$V_{pore} = V_{pore_{,ol}}\left(1 - C_{drug,0}\right) + V_{pore,drug}C_{drug,0}, \#\,(A.10)$$

$$V_{pore,drug} = 1 - \frac{C_{drug}}{C_{drug,0}}, \#(A.11)$$

$$V_{pore,ol} = \frac{R_{ol} - (C_{ol} - C_{ol,0})}{C_{e0}}, \#(A.12)$$

The model has four independent variables: R_s, R_{ol}, C_{drug} and C_{ol}. which are governed by (A-1), (A-3), (A-6), (A-7). They can be integrated numerically (using finite element method for complex shape for example). Once they are integrated, normalised mass loss is calculated by the difference between R_{ol} and C_{ol} divided by C_{eo}. Drug release amount is calculated by the integral of mole concentration of drug.

The number-averaged molecular weight is given by

$$M_n = \frac{\left(1 - \alpha\left(\frac{R_s}{C_{e0}}\right)^{\beta}\right)M_{n0}}{1 + N_{dp0}\left(\frac{R_s}{C_{e0}} - \frac{\alpha}{m}\left(\frac{R_s}{C_{e0}}\right)^{\beta}\right)}. \qquad \#\,(A.13)$$

Where N_{dp0} is the initial average degree of polymerisation of the polymer.

Appendix B. Numerical Details and Model Parameters

A finite difference scheme was used to solve Eqns. (A.1) - (A.13). As diffusion occurred only in one dimension in all the experiments conducted, all cases were simplified as one-dimensional problems with corresponding symmetry conditions. k_1, $k_2^{'}$, $k_3^{'}$ and D_0 were calibrated against the experimental data [14] with MATLAB R2022a (MathWorks, USA), whilst the values for the rest model parameters were provided in Refs. [14][15]. The values of model parameters for all cases are presented in Table B. The temporal evolutions of normalised average molecular weight (M_n normliased by its initial value M_{no} and averaged over the volume) and normalised mass loss (normliased by C_{e0} and averaged over the volume), numerically calculated by solving Eqns. (A.6)-(A.10) in MAT-

LAB for Cases A and B, are plotted in **Figure** 1. It should be noted that the sample with the thickness of 2 mm (Case B) fractured into pieces at around 30 weeks [8]. Consequently, the predictions made in this study could not be applied to this condition and, thus, its last data point for normalised mass loss was excluded. The temporal evolutions of normalised average molecular weight and drug release (normliased by f_{drug} and averaged over the volume) for Case C-E are plotted in **Figure 2**. Apparently, the aforementioned mathematical model with empirical equations cannot fully match the actual degradation behaviour of polymer. Our purpose here is to demonstrate that it can nevertheless be used as a starting point to train ANNs to capture the observed behaviour.

Table B: Values of model parameters in Eqns. (A.1)-(A.10) for Cases A-E.

Material parameters	Case A-B	Case C-E
M_o (g mol^{-1})	72	65
m	4	4
α	28	0.4
β	2	1
C_{eo} (mol m^{-3})	17300	20615
M_{no} (g mol^{-1})	30000	12500
Empirical kinetic parameters	Case A-B	Case C-E
K_1 (week^{-1})	1.4×10^{-4}	8×10^{-7}
$k_2^{'}$ ([mol^{-1} m^3]$^{0.5}$ week^{-1})	0.000475	1×10^{-7}
$k_3^{'}$ ([mol^{-1} m^3]$^{0.5}$ week^{-1})	-	0.004
D_0 (m^3 week^{-1})	8×10^{-7}	1×10^{-6}
D_{pore} (m^3 week^{-1})	8×10^{-4}	1×10^{-3}
D_{drug0} (m^3 week^{-1})	-	6×10^{-14}
$D_{drug0pore}$ (m^3 week^{-1})	-	6×10^{-11}

References

1. ZA Ali, R Gao, T Kimura, Y Onuma, DJ Kereiakes, SG Ellis, B Chevalier, et al. (2018) Three-year outcomes with the Absorb bioresorbable scaffold: Individual-patient-data meta--analysis from the ABSORB randomized trials, Circulation, 137: 464-79.

2. M Maintz, C Tourbier, M de Wild, PC Cattin, M Beyer, D Seiler, et al. (2024) Patient-specific implants made of 3D printed bioresorbable polymers at the point-of-care: material, technology, and scope of surgical application, 3D Print. Med., 10: 13.

3. M Cho, JK Han, J Suh, JJ Kim, JR Ryu, IS Min, et al. (2024) Fully bioresorbable hybrid opto-electronic neural implant system for simultaneous electrophysiological recording and optogenetic stimulation, Nat. Commun., 15: 2000.

4. ARM Sikkander, H Yadav, M Meena, VV Lakshmi (2024) A Review of Advances in the Development of Bioresorbable Nano Stents: Part (II). J. Chem. Rev. 6: 304-30.

5. Y Wang, J Pan X. Han, C Sinka, L Ding (2008) A phenomenological model for the degradation of biodegradable polymers, Biomaterials. 29L 3393-401.

6. A Gleadall, J Pan, MA Kruft, M Kellomäki (2014) Degradation mechanisms of bioresorbable polyesters. Part 1. Effects of random scission, end scission and autocatalysis, Acta Biomaterialia. 10: 2223-32.

7. K Sevim, J Pan (2018) A model for hydrolytic degradation and erosion of biodegradable polymers, Acta Biomaterialia. 66: 192-9.

8. K Sevim, J Pan (2014) A mechanistic model for acidic drug release using microspheres made of PLGA 50:50, Molecular Pharmaceutics. 13: 2729-35.

9. AR Abaei, CJ Shine, TJ Vaughan, W Ronan (2024) An integrated mechanical degradation model to explore the mechanical response of a bioresorbable polymeric scaffold, Journal of the Mechanical Behavior of Biomedical Materials. 152: 106419.

10. J Pan (2015) Modelling Degradation of bioresorbable Polymeric Medical Devices, Woodhead Publishing, Cambridge.

11. W Niu, J Pan (2020) A model of polymer degradation and erosion for finite element analysis of bioresorbable implants, Journal of the Mechanical Behavior of Biomedical Materials. 112: 104022

12. Y Hua, Y Su, et al. (2021) Poly(lactic-co-glycolic acid) microsphere production based on quality by design: a review, Drug Delivery. 28: 1342-55.

13. A Li, R Chen, AB Farimani, YJ Zhang (2020) Reaction diffusion system prediction based on convolutional neural network, Scientific Reports. 10: 3894.

14. I Grizzi, H Garreau, S Li, M Vert (1995) Hydrolytic degradation of devices based on poly(dl-lactic acid) size-dependence. Biomaterials. 16: 305-11.

15. JF Fitzgerald, OI Corrigan (1996) Investigation of the mechanisms governing the release of levamisole from poly-lactide-co-glycolide delivery systems, Journal of Controlled Release. 125-32.

16. R He, V Ghantasala, P Polak, B Saleem, J Pan (2024) Physics-based neural network as constitutive law for finite element analysis of sintering, Ceramics International. 50: 37291-9.

17. DG Rizik, JB Hermiller, CA Simonton, KJ Klassen, DJ Kereiakes (2017) Bioresorbable vascular scaffolds for the treatment of coronary artery disease: What have we learned from randomized-controlled clinical trials? Coron. Artery Dis. 28: 77-89.

18. S Cassese, RA Byrne, G Ndrepepa, S Kufner, J Wiebe, J Repp et al. (2016) Everolimus-eluting bioresorbable vascular scaffolds versus everolimus-eluting metallic stents: A meta-analysis of randomised controlled trials. Lancet. 387: 537-44.

19. K Linka, M Hillgärtner, KP Abdolazizi, RC Aydin, M Itskov, CJ Cyron (2021) Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning. J. Comput. Phys. 429: 110010.