



Ensemble of heterogeneous flexible neural tree for the approximation and feature-selection of Poly (Lactic-co-glycolic Acid) micro-and nanoparticle

Conference or Workshop Item

Accepted Version

Ojha, V. ORCID: <https://orcid.org/0000-0002-9256-1192>, Abraham, A. and Snasel, V. (2016) Ensemble of heterogeneous flexible neural tree for the approximation and feature-selection of Poly (Lactic-co-glycolic Acid) micro-and nanoparticle. In: Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015, Sep 9, 2015 - Sep 11, 2015, Paris - Villejuif, France, pp. 155-165. doi: https://doi.org/10.1007/978-3-319-29504-6_16 Available at <http://centaur.reading.ac.uk/93556/>

It is advisable to refer to the publisher's version if you intend to cite from the work. See [Guidance on citing](#).

Published version at: http://dx.doi.org/10.1007/978-3-319-29504-6_16

To link to this article DOI: http://dx.doi.org/10.1007/978-3-319-29504-6_16

Publisher: Springer Science \mathplus Business Media

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the [End User Agreement](#).

www.reading.ac.uk/centaur

CentAUR

Central Archive at the University of Reading

Reading's research outputs online

Ensemble of Heterogeneous Flexible Neural Tree for the approximation and feature-selection of poly (lactic-co-glycolic acid) micro- and nanoparticle

Varun Kumar Ojha¹, Ajith Abraham¹, Vaclav Snasel¹

¹IT4Innovations, VŠB Technical University of Ostrava, Ostrava, Czech Republic
{varun.kumar.ojha,ajith.abraham,vaclav.snasel}@vsb.cz

Abstract. In this work, we used an adaptive feature-selection and function approximation model, called, flexible neural tree (FNT) for predicting Poly (lactic-co-glycolic acid) (PLGA) micro- and nanoparticle's dissolution-rates that bears significant role in the pharmaceutical, medical, and drug manufacturing industries. Several factor influences PLGA nanoparticles dissolution-rate prediction. FNT model enable us to deal with feature-selection and prediction simultaneously. However, a single FNT model may or may not offer a generalized solution. Hence, to build a generalized model, we used an ensemble of FNTs. In this work, we have provided a comprehensive study for examining the most significant (influencing) features that influences dissolution rate prediction.

Keywords: poly (lactic-co-glycolic acid) (PLGA) micro- and nanoparticle, flexible neural tree, function approximation, feature selection.

1 Introduction

Application domains, such as chemical, pharmaceutical, medical, biology, etc., yields a large volume of data with a large number of features. Researchers often use computational intelligence tools for data analysis, primarily, for discovering the relationship between independent and dependent variables. Traditional data-driven computational intelligence method are insufficient in dealing with such volume of data with high accuracy. Mostly, a single function-approximation model may not be able to offer the most general solution to a problem. In this work, we solve a pharmaceutical industry problem that sought for a computational model that can predict the dissolution-rate profile of poly (lactic-co-glycolic acid) (PLGA) micro-and nanoparticles. PLGA dissolution profile prediction is a complex problem in terms of the number of feature that governs its dissolution-rate profile. A thorough examination of the academic literature provides us 300 potential factors that may influence the dissolution-rate of the PLGA protein particles [1] [3].

In this work, our approach was to find the most significant features that will govern the dissolution profile and to construct an approximation model with high accuracy

without the loss of generality. The significance of the features selection is not limited only to finding most significant features, but it also reduces the computational complexity of the problem and significantly contributes in improving model’s computational speed, predictability, and generalization ability. There are several techniques available for the pre-selection of the most probable features [2]. However, in this work, we presents a computational tool, namely, heterogeneous flexible neural tree-(HFNT) that offers automatic features selection and function approximation. Interestingly, HFNT also make sure diversity in the feature space that makes it an adaptive feature selector. In general, combined knowledge of many experts (predictors) offers better solution in terms of reliability and generality than that of a single expert. So, an ensemble of several HFNT would offer most general solution.

We provides a comprehensive discussion on the drug release problem and dataset collection mechanisms that tells the significance of macro-and nanoparticle dissolution-rate profile prediction (section 2.1). In the following selection, we will offer a discussion on HFNT and ensemble methods. Finally, we discuss the experimental results followed by conclusion.

2 Methodology

2.1 The PLGA dissolution-rate prediction problem

PLGA micro- and nanoparticles plays a significant role in the medical application and toxicity evaluation of PLGA-based multi-particulate dosages [13]. PLGA micro-particles are important diluents used to produce drugs in their correct dosage form. Apart from playing the role as a filler, PLGA as an excipient, and alongside pharmaceutical APIs, plays other crucial roles in various ways. It helps in the dissolution of drugs, thus increasing the absorbability and solubility of drugs [14], [15]. It helps in pharmaceutical manufacturing processes by improving API powders’ flow ability and non-stickiness.

The dataset collected from various academic literature contains 300 input features categorized into four groups, including protein descriptor, plasticizer, formulation characteristics, and emulsifier. Table 1 provides a detailed description of the dataset.

Table 1. The PLGA dataset description

#	Group name	# features	Importance
1	Protein descriptors	85	Describes the type of molecules and proteins used
2	Formulation characteristics	17	Describe the molecular properties such as molecular weight, particle size, etc.
3	Plasticizer	98	Describe the properties such as fluidity of the material used
4	Emulsifier	99	Describe the properties of stabilizing/increase the pharmaceutical product life
5	Time in days	1	Time taken to dissolve
6	% of molecules dissolved	1	PLGA micro-nanoparticle dissolution-rate

2.2 Background Study

In the scope of this study, our focus was on PLGA dissolution properties and drug release rate. Szlęk *et al.* [3] and Fredenberg *et al.* [4] described two mechanisms: diffusion and degradation/erosion that is mainly governs the drug release from the PLGA matrix. Several factors influencing the diffusion and degradation rates of PLGAs described by Kang *et al.* [5], [6] Blanco and Alonso [7] and Mainardes and Evangelista [8] includes pore diameters, matrix active pharmaceutical ingredient (API) interactions, API-API interactions, and the formulation composition. Szlęk *et al.* [3] developed a predictive model to describe the underlying relationship between those influencing factors on the drug's release profile, they focused on feature selection, artificial neural network, and genetic programming approaches to come up with a suitable prediction model. In the past, several mathematical models, including the Monte Carlo and cellular automata microscopic models were proposed by Zygorakis and Markenscoff [9] and Gopferich [10]. A partial differential equation model was proposed by Siepmann *et al.* [11] to address the influence of underlying PLGA properties on the drug's release rate/protein dissolution. Ojha *et al.* [12] used a pool of several trained predictors and made an ensemble to get a model with high predictability. However, they had to use separate techniques for the feature-selection beforehand to make approximation models, in contrast to that, in this work, we proposed a tool for function-approximation and feature-selection simultaneously.

2.3 Flexible Neural Tree Approach

For a dataset with n many independent variables X and a dependent variable Y , an approximation model tries finds relationship between them. Moreover, it tries to find unknown parameter θ such that root mean square error (RMSE) between models' output \hat{Y} and actual output Y be zero. Therefore, we may write RMSE as:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}, \quad (1)$$

where N is number of examples.

A wide range application accepts artificial neural network (ANN) as most convenient tool for the approximation [16]. Thus, makes it a universal approximator. ANN performance heavily relies on its structure, parameters, and activation-functions [16] optimization. Researchers have investigated various ways in the past to optimize the individual components of ANN using evolutionary procedure [17, 18]. Chen *et al.* [19] proposed flexible neural tree (FNT) that addressed ANN optimization in all its components including structure and parameters. And, does an automatic feature selection. FNT was conceptualized around a multi-layered feed-forward neural network to build a tree based model, where network structure and parameters were optimized by using meta-heuristic optimization algorithms (the nature inspired stochastic algorithms for function optimization).

An FNT can be defined as a set of function-nodes and terminals, where the function-node indicates a computational node and terminals indicated a set of all input

features. The function instruction set F and terminal instruction set T for generating an FNT model are described as follows:

$$S = F \cup T = \{+_2, +_3, +_4, \dots, +_N\} \cup \{x_1, x_2, \dots, x_n\}, \quad (2)$$

where $+_i$ ($i = 2, 3, \dots, n$) indicates that a function-node can take i arguments, whereas, the leaf node (terminal node) receives no arguments. Figure 1 illustrates a function-node/computational-node of an FNT.

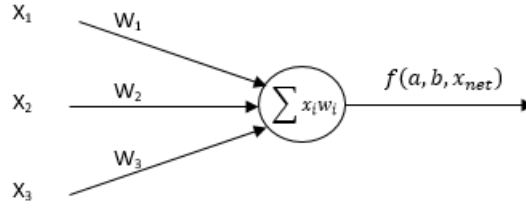


Fig 1. A computational node of a flexible neural tree

In Figure 1, the computational node $+_i$ receives i inputs through i connection weights (random real values) and two adjustable parameters/arguments a_i and b_i of the squashing (transfer) function, that limits the total output of the function-node within a certain range. A transfer-function used at the function-node is:

$$f(a_i, b_i, net_n) = e^{-\left(\frac{net_n - a_i}{b_i}\right)}, \quad (3)$$

where net_n is the net input to the i th function-node also known as excitation of the node computed as:

$$net_n = \sum_{j=1}^n w_j x_j, \quad (4)$$

where $j = 1, 2, 3 \dots$ is the input to the i th node. Therefore, the output of the i th node is given as:

$$out_n = f(a_i, b_i, net_n) = e^{-\left(\frac{net_n - a_i}{b_i}\right)}, \quad (5)$$

Figure 2 illustrates an example of a typical FNT. The root node of the FNT given in Figure 2 indicates the output of the entire tree-based model. The leaf nodes of the tree are indicating the selected input feature and the edges of the tree indicate the underlying parameters (or the weights) of the model. We can choose various kinds of activation-function at the function-nodes of an FNT and genetic evaluation give various structure at various instances. So, FNTs with heterogeneity in its function-nodes and structure is now on, called, heterogeneous flexible neural trees (HFNTs). Fitness of a HFNT is computed according to (1). The root node of a HFNT returns the output of the entire model.

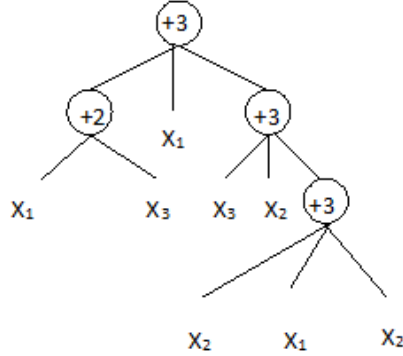


Figure 2. A typical FNT with instruction set $F = \{+2, +3, \}$ and $T = \{x_1, x_2, x_3\}$

Meta-heuristic framework for HFNT optimization: We can classify HFNT optimization in two different parts:

- (1) The optimization of the HFNT structure, in other words, the first approach is to find a near optimum tree using genetic programming that works on a genetic population and uses operators like crossover and mutation to evolve new generation [26].
- (2) The optimization of the parameters of the tree, i.e., the optimization of the edges and the arguments of the activation function used at the function-nodes. Swarm based meta-heuristic were used for optimizing parameters, where swarm-based meta-heuristics are the algorithms that are inspired from the foraging behavior of the swarm such as group of fish, birds, bees, etc. [27]

2.4 Ensemble of HFNT

A collective decision with consensus of many candidates is better than decision of an Individual. Hence, ensemble of many models (predictors) may offer the most general solution to a problem [20]. There are two components in ensemble system [21] [22] construction:

- (1) Construction of as diverse and as accurate models as possible. To construct diverse and accurate models following techniques may be used: (a) training models with different sets of data, like bagging algorithm [28]; (b) training models with different set of input features, Random Sub-space algorithm; and (c) training models with different set of parameters.
- (2) Combining the models using a combination rules. Once many models constructed with high diversity and accuracy, then we need to combine them for a collective decision. We used weighted mean combination method, where the weights for the models were computed by using meta-heuristic algorithm.

3 Experimental Set-up and Results

We conducted our experiment using a platform independent software tool that we developed for realizing FNT. We used the develop software tool for processing of PLGA dataset that has 300 features and 754 examples. We used training parameters setting as mentioned in Table 2.

We conducted several experiments with the parameter settings mentioned in Table 2. Since, the computation model mentioned is stochastic in nature, each instance of experiment offer distinct results in terms of accuracy and future selection. The Accuracy, in other words, fitness of an approximation model was measured in terms RMSE given in (1). However, the correlation that tells relationship between two variables (in this case, the actual output, and the models' output) reveals the quality of the constructed model.

We have selected four highly accurate and divers models for making ensemble. The obtained results is provided in Table 3.

Table 2. The parameters setting of the HFNT software tool.

#	Parameter Name	Parameter Utility	Values
1	Tree Height	Maximum number of level of FNT	5
2	Tree Arity	Maximum number of siblings a node.	10
3	Tree Node Type	Indicates the type of activation at nodes.	Gaussian
4	GP Population	Number of candidates in genetic population.	30
5	Mutation probability	The probability that a candidate will be mutated in a genetic programming.	0.4
6	Crossover probability	The probability that candidates will take part in crossover operation.	0.5
7	elitism	The probability that a candidate will propagate to next generation as it is.	0.1
8	Tournament Size	It indicate the size of the pool used for the selection of the candidates.	15
9	MH Algorithm Population	The initial size of the swarm (population).	50
10	MH Algorithm Node Range	It defines the search space of the transfer-function arguments.	[0,1]
11	MH Algorithm Edge Range	It defines the search space for the edges (weights) of tree.	[-1.0,1.0]
13	Maximum Structure Iteration	Maximum number of generation of genetic programming	100000
14	Maximum Parameter Iteration	Maximum number of evaluation of parameter optimization	10000

Table 3. RMSE and r of the constructed models and ensemble based on 10 cross validation.

Exp.	Training		Test		Ensemble weights
	RMSE	r	RMSE	r	
1	12.885	0.908	12.741	0.909	0.355686
2	12.907	0.908	13.248	0.903	0.319278
3	13.855	0.897	13.776	0.897	0.259922
4	14.599	0.881	14.374	0.884	0.085882
Ensemble	-	-	11.541	0.928	-

We select model (second in Table 3) for comparing with the results available in literature. Our second model selects fifteen features and offer a cross validation RMSE 13.248. A comprehensive list of models available in literature for PLGA prediction is given in Table 4.

Table 4. Best models constructed for PLGA prediction problem (cross validation results).

#	RMSE	Features	Model	Literature
1	13.248	15	HFNT	Current Work
2	13.34	15	REP Tree [24]	Ojha <i>et al.</i> [12]
3	14.3	17	MLP [16]	Szłęk <i>et al.</i> [3]
4	14.88	15	GP Regression [23]	Ojha <i>et al.</i> [12]
5	15.2	15	MLP [16]	Ojha <i>et al.</i> [12]
6	15.4	11	MLP [16]	Szłęk <i>et al.</i> [3]

Main focus of the present experiment as promised at beginning was to find-out the significant features and the construction of accurate model. In previous part of results section, we have presented accurate model. However, we were resistant to demonstrate significant features. To obtain significant features, we repeated our experiment 20 times. We listed all the features selected by each instant of experiment according to their probability of selection. In Table 5, we have provided a list of 15 such features whose probability of selection were obtained higher than 0.2. Similarly, Table 6 was realized. In Table 6, we have presented a picture of overall group probability. The probability of feature selected from particular group was computed by averaging the probabilities of all the features those were fall under that particular group.

Table 5. Most significant feature (according to probability of selection). Individual assessment.

#	Feature	Index	Abbreviation	Probability
1	Time Days	299	TD	0.94
2	Prod method	100	PM	0.83
3	PVA conc. inner phase	88	PVA	0.78
4	Ring atom count	110	RAC	0.61
5	Heteroaliphatic ring count	23	HIRC	0.50
6	Aliphatic bond count	104	ABC	0.44
7	Diss. add	98	DA	0.44
8	pH 11 msdon	195	PH11MD	0.39
9	pH 12 msacc	181	PH12MC	0.39
10	Ring count	23	RC	0.39
11	a(yy)	119	AYY	0.28
12	Chain bond count	213	CBC	0.28
13	Diss. add conc.	99	DAC	0.28
14	Fragment count	133	FC	0.28
15	Aromatic ring count	24	ARC	0.22

Table 6. Most significant feature (Group wise assessment)

Sl. No.	Group name	# features	Probability	Selected Feature	# Most Significant Features	
					#	Name
1	Protein descriptors	85	0.15	17	3	HRC, RC, ARC
2	Formulation characteristics	17	0.28	10	4	PM, PVA, DA, DAC
3	Plasticizer	98	0.16	29	6	RAC, ABC, AYY, FC, PH11MD, PH12MC,
4	Emulsifier	99	0.1	38	1	CBC
5	Time in days	1	0.94	1	1	TD

4 Discussion

The aim of PLGA dissolution-rate prediction experiment was to find the significant variables that governs the prediction rate and to create a model for realizing the PLGA prediction profile. The results obtained from the experiment suggests significant improvement over the earlier models developed [3] [12]. Our current experiment provides an insight of the PLGA dissolution rate prediction. We have discovered a list of most significant features by computing their probability of selection using our model (higher the probability of selection, higher the significance on prediction). The probability was computed by examining the features those were repeatedly selected by the distinct experiments. The results of ensemble of distinct predictors (models) helped in achieving high accuracy, whereas, the individual models were using their own set of features. Hence, the ensemble model was using a union of all features sets by all models. However, the second model (see Table 3) gave us a better result than the earlier models reported in [3] [12]. The developed tool was hence able to address the problem related to the prediction modeling efficiently.

5 Conclusion

Computational Intelligent tools are widely used for data analysis in the industries such pharmaceutical, medical, chemistry, etc. In this article, a function approximation and feature-selection tool was used for the identification of the significant features that govern the prediction of PLGA micro-nanoparticles. The computational intelligence tool mentioned in this paper was a tree-based implementation of neural network that provides optimum network structure and parameters. The entire model was optimized using meta-heuristic algorithms. Hence, it helps in creation of diverse (heterogeneous) models in the ensemble system. The ensemble of heterogeneous models offered better result than that of all the earlier models. However, the size (complexity) of the tree was a concern. A multi-objective approach may be useful in dealing with the conflicting objectives such as complexity and accuracy.

Acknowledgment

This work was supported by the IPROCUM Marie Curie Initial Training Network, funded through the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007–2013/, under REA grant agreement number 316555.

References

1. Astete CE, Sabliov CM (2006) Synthesis and characterization of PLGA nanoparticles. *Journal of Biomaterials Science, Polymer Edition*. 17(3):247–289.
2. van der Maaten LJ, Postma EO, van den Herik HJ. Dimensionality Reduction: A comparative review. Technical Report TiCC TR 2009-005.
3. Szlęk J, Paclawski A, Lau R, Jachowicz R, Mendyk A (2013) Heuristic modeling of macromolecule release from PLGA microspheres. *International Journal of Nanomedicine*. 8(1):4601–4611.
4. Fredenberg S, Wahlgren M, Reslow M, Axelsson A (2011) The mechanisms of drug release in poly (lactic-co-glycolic acid)-based drug delivery systems: a review. *International Journal of Pharmaceutics*. 415(1–2):34–52.
5. Kang J, Schwendeman SP (2007) Pore closing and opening in biodegradable polymers and their effect on the controlled release of proteins. *Molecular Pharmaceutics*. 4(1):104–118.
6. Kang J, Lambert O, Ausborn M, Schwendeman SP (2008) Stability of proteins encapsulated in injectable and biodegradable poly(lactide-co-glycolide)-glucose millicylinders. *International Journal of Pharmaceutics*. 357(1):235–243.
7. Blanco MD, Alonso MJ. Development and characterization of proteinloaded poly(lactide-co-glycolide) nanospheres. *European Journal of Pharmaceutics and Biopharmaceutics*. 1997; 43(3):287–294.
8. Mainardes RM, Evangelista RC (2005) PLGA nanoparticles containing praziquantel: effect of formulation variables on size distribution. *International Journal of Pharmaceutics*. 290(1–2):137–144.
9. Zygourakis K, Markenscoff PA (1996) Computer-aided design of bioerodible devices with optimal release characteristics: a cellular automata approach. *Biomaterials*. 17(2):125–135.
10. Gopferich A (1996) Mechanisms of polymer degradation and erosion. *Biomaterials*. 17(2):103–114.
11. Siepmann J, Faisant N, Benoit JP (2002) A new mathematical model quantifying drug release from bioerodible microparticles using Monte Carlo simulations. *Pharmaceutical Research*. 19(12):1885–1893.
12. Ojha VK, Jackowski K, Abraham A, Snášel V (2015) Dimensionality Reduction, and Function Approximation of Poly(lactic-Co-Glycolic Acid) Micro- and Nanoparticle Dissolution Rate. *International Journal of Nanomedicine* 10: 1119–1129.

13. Langer R, Tirrell DA (2004) Designing materials for biology and medicine. *Nature*. 428:487–492.
14. Brodbeck KJ, DesNoyer JR, McHugh AJ (1999) Phase inversion dynamics of PLGA solutions related to drug delivery. Part II. The role of solution thermodynamics and bath-side mass transfer. *Journal of Controlled Release*. 62(3):333–344.
15. Makadia HK, Siegel SJ (2011) Poly Lactic-co-Glycolic Acid (PLGA) as biodegradable controlled drug delivery carrier. *Polymers (Basel)*; 3(3):1377–1397.
16. Haykin S (1994) *Neural Networks: A Comprehensive Foundation*. 1st ed. Upper Saddle River, NJ: Prentice Hall PRT.
17. Goldberg DE (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning*. Boston, MA: Addison-Wesley.
18. Yao X (1999) Evolving artificial neural networks. *Proceedings of the IEEE* 87(9): 1423-1447.
19. Chen, Y, Yang B, Dong J, and Abraham A (2005) Time-series forecasting using flexible neural tree model. *Information Sciences* 174 (3): 219-235.
20. Dietterich, Thomas G (2000) Ensemble methods in machine learning. *Multiple classifier systems*. Springer Berlin Heidelberg, p 1-15.
21. Mendes-Moreira, J., Soares, C., Jorge, A. M., et al (2012) Ensemble approaches for regression: A survey. *ACM Computing Surveys (CSUR)*, 45(1), 10.
22. Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and systems magazine*, 6(3): 21-45.
23. Rasmussen CE, Williams CK (2005) *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. Cambridge, MA: MIT Press.
24. Quinlan JR (1987) Simplifying decision trees. *International Journal of Man-Machine Studies*. 27(3):221–234.
25. Smola AJ, Schölkopf B (1998) *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA: MIT Press.
26. Goldberg, David E., and John H. Holland (1988) Genetic algorithms and machine learning. *Machine Learning* 3(2): 95-99.
27. Kennedy, J (2010) Particle swarm optimization. *Encyclopedia of Machine Learning*. Springer US p 760-766.
28. Breiman, L (1996). Bagging predictors. *Machine Learning* 24.2: 123-140.