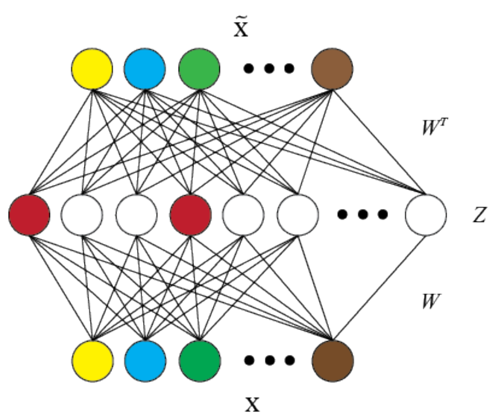
Deep Learning as a Tool to Predict Flow Patterns in Two-Phase Flow

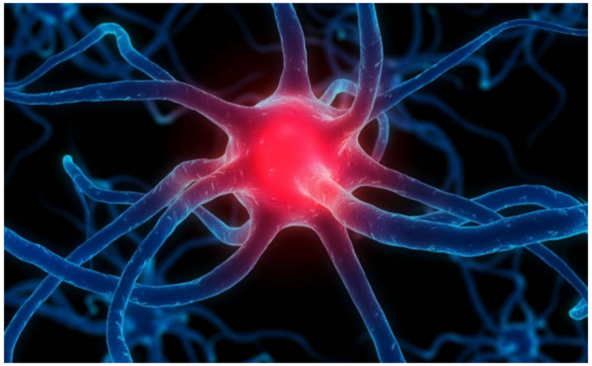
M. Ezzatabadipour
P.Singh
M. Robinson
C. Torres
P. Guillén
pgrondon@uh.edu
Center for Advanced Computing and Data Systems
University of Houston

INTRODUCTION

In order to better model complex real-word data such as multiphase flow, one approach is to develop:

- **Pattern recognition techniques**
- **Robust features**
- **Deep learning methods - MLP**
- **Predict flow patterns**
- **From fluid properties and pipe conditions**





INTRODUCTION

- Human information processing mechanism (e.g. vision and speech) suggests the need of deep architectures for extracting complex structure and building internal representation from rich sensory inputs

- It is natural to believe that the state of the art can be advanced in processing these types of media signals if efficient and effective deep learning algorithms are developed

- Deep architectures are composed of many layers of nonlinear processing stages, where each lower layer's outputs are fed to its immediate higher layer as the input

- The concept of deep learning originated from artificial neural network research. Multilayer perceptron with many hidden layers is a good example of the models with deep architectures

- Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction

INTRODUCTION

- The term flow pattern refers to the spatial distribution of the phases, which occur during gas-liquid two-phase flow in pipes

- When gases and liquids flow simultaneously in a pipe, the two phases can distribute themselves in a variety of flow configurations

- The flow configurations differ from each other in the interface distribution, resulting in different flow characteristics

- Determination of flow patterns is a fundamental problem in two-phase flow analysis. Thus, knowledge of the existing flow pattern can help the industry carry out a better design of two-phase flow systems

- There is not agreement in the number of flow patterns in two-phase flow due to overlapping and characterization subjectivity, especially at the transition zones

- [Shoham 2006] attempted to summarize the main flow patterns for all inclination angles as Dispersed bubble, Bubble, Slug, Churn, Annular and Stratified (smooth and wavy). The flow patterns depend on parameters such as pipe inclination and diameter, physical properties of the phases, and their superficial velocities

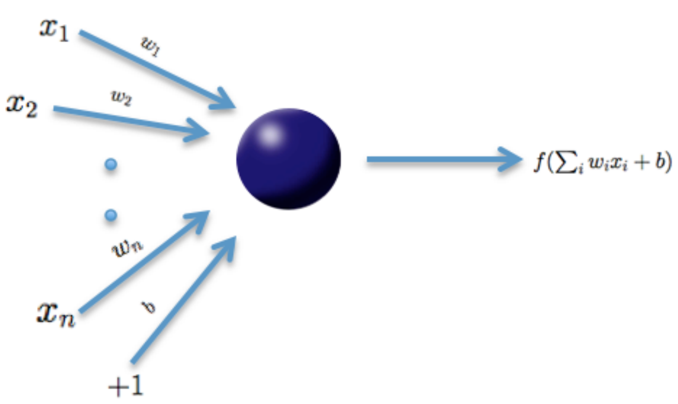
MATERIALS and METHODS

Deep Learning: Multilayer Perceptron (MLP)

- **There are several theoretical frameworks for deep learning, and here we summarize the feedforward architecture used by H20.**

- **Multilayer perceptron (MLP) are feed-forward neural networks with architecture composed of the input layer, the hidden layer and the output layer.**

- **Each layer is formed from small units known as neurons. Neurons in the input layer receive the input signals X and distribute them forward to the rest of the network. In the next layers, each neuron receives a signal, which is a weighted sum of the outputs of the nodes in the previous layer. Inside each neuron, an activation function is used to control the input.**



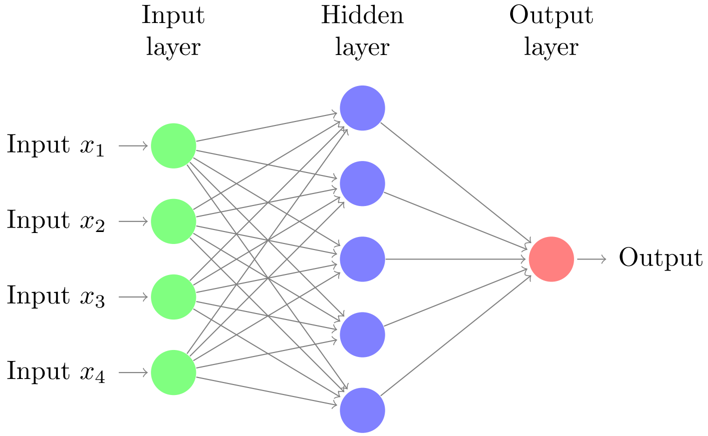
MATERIALS and METHODS

Deep Learning: Multilayer Perceptron (MLP)

- **Such a network determines a non-linear mapping from an input vector to the output vector, parameterized by a set of network weights, which are referred to as the vector of weights W.**

- **The first step in approximating the weight parameters of the model is finding the appropriate architecture of the MLP, where the architecture is characterized by the number of hidden units, the type of activation function, as well as the number of input and output variables.**

- **The second step estimates the weight parameters using the training set. Training estimates the weight vector W to ensure that the output is as close to the target vector as possible.**



MATERIALS and METHODS

Deep Learning: Multilayer Perceptron (MLP)

- **The purpose of this training is to learn the multilayer architectures by simple stochastic gradient descent. The backpropagation procedure to compute the gradient of an objective function with respect to the weights of a multilayer stack of modules is nothing more than a practical application of the chain rule for derivatives.**

- **The key insight is that the gradient of the objective with respect to the input of a module can be computed by working backward from the gradient with respect to the output of that module. The backpropagation equation can be applied repeatedly to propagate gradients through all modules, starting from the output at the top (where the network produces its prediction) all the way to the bottom (where the external input is fed)**

- **Multilayer feedwoard neural networks consist of many layers of interconnected neuron units, starting with an input layer to match the feature space, followed by multiple layers of nonlinearity, and ending with a linear regression or classification layer to match the output space.**

MATERIALS and METHODS

Deep Learning: Multilayer Perceptron (MLP)

- **The weights linking neurons and biases with other neurons fully determine the output of the entire network, and finally learning occurs when these weights are adapted to minimize the error on labeled training data.**

- **To go from one layer to the next, the weighted sum of their inputs from the previous layer pass the result through a non-linear function.**

- **At the present, the most popular non-linear function is the rectified linear unit (ReLU).**

MATERIALS and METHODS

Database

- **A flow pattern experimental data base was collected, which consists of the most relevant studies developed in the area.**

- **Specifically for this study, the data set from [Shoham 1982] was selected among the available sets due to its large number of data points (5676), range in inclination angle (-90° to 90°), two pipe diameters (ID=1in and 2in), and the wide range of flow patterns observed for all pipe inclination angles.**

- **The flow patterns considered in this study are: Annular (A), Bubble (B), Dispersed bubble (DB), Intermittent (I), Stratified smooth (SS) and Stratified wavy (SW). The Intermittent flow pattern considers Slug (SL) and Churn (CH) flow pattern combined.**

- **In order to analyze the performance of the algorithm, three tests are proposed: Test 1 considers all the flow patterns proposed; Test 2 combines the SS and SW data points into stratified flow ST (ST = SS + SW); finally Test 3 combines the segregated flow patterns (ST + A) and the dispersed flow patterns (DB + B).**

RESULTS

- **Table 1 shows the size of the architecture multilayer perceptron and parameters used on the experiments to evaluate the classification**

Variables	Parameters
Number of input neurons	9
Number of hidden layers	3
Hidden layer topology	(25,25,25)
Number of output neurons (classes)	6
Activation function	ReLU
Loss function	Mean Squared Error
Number of training epochs	1000000
l1 penalty weighting	0.00001
l2 penalty weighting	0.00001
n-fold cross-validation	10

In our approach, we train a MLP on a set of randomly selected features, approximately 60 %, extracted from the entire dataset, then approximately 20 % are used as the validation set, and approximately 20 % are used as the testing set for the 3 different tests

RESULTS

Table 2 shows the confusion matrix for the training data set for Test 1, predicting classes A, B, DB, I, SS, and SW. We can readily see the strong diagonal components. This means that our classifier is achieving little classification error. The confusion matrix's columns represent the output patterns predicted by Deep Learning while the rows represent the true class which is denoted here by each flow pattern.

Table 2: Training Data Confusion Matrix: Test 1

	A	B	DB	I	SS	SW	Error	Rate
A	617	0	0	4	0	0	0.0064412	4/621
B	0	76	0	0	0	0	0.0	0/76
DB	0	0	331	34	0	0	0.0931507	34/365
I	42	46	60	1629	0	6	0.0863713	154/1783
SS	0	0	0	56	14	10	0.825	66/180
SW	114	0	1	43	5	330	0.3306288	163/493
	773	122	392	1766	19	346	0.1231714	421/3418

RESULTS

The testing set is used to predict the variable Flow Pattern, which contains labels for each class (A, B, DB, I, SS, and SW), and a predictive accuracy of 83.87% for the different classes is obtained, the details of which are shown in Table 3.

Table 3: Confusion matrix for the cross-validation data set Test 1

	A	B	DB	I	SS	SW	Error	Rate
A	581	0	3	20	0	17	0.0644122	40/621
B	0	51	0	25	0	0	0.3289474	25/76
DB	2	0	310	50	0	3	0.1506849	55/365
I	78	29	65	1474	67	70	0.1733034	309/1783
SS	0	0	0	5	67	8	0.1625	13/80
SW	94	0	2	23	26	348	0.2941176	145/493
	755	80	380	1597	160	446	0.1717379	587/3418

RESULTS

Table 4 shows the confusion matrix on train data for Test 2, predicting classes A, B, DB, I and ST. Similar to Test 1 we can see the strong diagonal components, and the classifier has small classification error. The testing set is used to predict the variable Flow Pattern, which contains labels for each class (A, B, DB, I, and ST), and we achieve a predictive accuracy of 83.34%. The details are shown in Table 5.

Table 4: Training Data Confusion Matrix: Test 2

	A	B	DB	I	ST	Error	Rate
A	605	0	0	6	10	0.0257649	16/621
B	0	76	0	0	0	0.0	0/76
DB	0	0	350	14	1	0.2	15/365
I	69	40	109	1433	132	0.1962984	350/1783
ST	93	0	1	1	478	0.1657941	95/573
	767	116	460	1454	621	0.1392627	476/3418

RESULTS

Table 5: Confusion matrix for the cross-validation data set Test 2

	A	B	DB	I	ST	Error	Rate
A	574	0	2	21	24	0.0756844	47/621
B	0	35	1	40	0	0.5394747	41/76
DB	0	0	309	53	3	0.1534247	56/365
I	82	15	94	1436	156	0.1946158	347/1783
ST	98	0	4	28	443	0.2268761	130/573
	754	50	410	1578	626	0.1816852	621/3418

RESULTS

Table 6 shows the confusion matrix for the training data set for Test 3, predicting the classes Intermittent, Dispersed, and Segregate. We can readily see the strong diagonal components with the classifier achieving little classification error. The testing set is used to predict the variable flow pattern, which contains labels for each class (Intermittent, Dispersed, and Segregate) with a predictive accuracy of 85.97%, the details of which are shown in Table 7

Table 6: Training Data Confusion Matrix: Test 3

	Intermittent	Dispersed	Segregate	Error	Rate
Intermittent	1419	63	301	0.2041503	364/1783
Dispersed	64	295	6	0.1917808	70/365
Segregate	85	2	1183	0.0685039	87/1270
	1490	420	1508	0.1524283	521/3418

Table 7: Confusion matrix for the cross-validation data set Test 3

	Intermittent	Dispersed	Segregate	Error	Rate
Intermittent	1469	70	224	0.1761077	314/1783
Dispersed	14	350	1	0.0410959	15/365
Segregate	7	0	1263	0.00055118	7/1270
	1490	420	1508	0.09833031	336/3418

RESULTS

Discussion

- A comparison between the predicted flow pattern and the experimental database considering the three data sets under study show low error and high classification accuracy.
- Results for Test 1 and Test 2 are very similar. Most of the failed predictions between the flow patterns can be attributed to the different criteria used by the different experimentalists to classify the flow patterns and their relationships.
- Finally an improvement is obtained for Test 3 by combining the segregated flow patterns (ST + A) and the dispersed flow patterns (DB + B). The prediction accuracy for this case increases to 85.97%. This is improvement is due to the clear and straightforward distinction between the two combined flow patterns [Shoham].
- The results for the deep learning approach for classification of two-phase flow pattern are encouraging.

CONCLUSIONS

- In this paper we proposed three types of data sets as input features and investigated the use of deep learning for the classification and prediction of two-phase flow, based on experimental data

- We proposed six types of input features, and a corresponding architecture to precisely predict flow patterns. First, we showed that the network can learn surprisingly well as using our chosen architecture and parameters allowed us to achieve high classification accuracy. Second, we showed that the network can classify the different flow patterns with high efficiency. Finally, we achieved high precision predicting different combinations of classes.

- Our experiments indicate that a deep learning approach, has the potential to capture flow patterns, which may boost the classification performance. These investigations could be further improved in future studies by carrying out more exhaustive searches for the parameters in the architectures. The result would be improved overall performance of these systems.

- Finally, deep learning can be used to predict flow patterns using pipe characteristics, fluid properties and superficial velocities of the two-phase flows. It outperforms results from previous studies