Machine Learning in Pipeline Inspection: Applications of supervised learning in non-destructive evaluation

Roberto H. HERRERA [1], Paul CHRISTENSEN [1], Adrianus ELVERS [1]

[1] UT Quality Inc. a RAE Energy Company; Edmonton, Canada
Phone: +1 780 930 2802; Fax: +1 780 484 5664; e-mail: h.herrera@rae.com, p.christensen@rae.com, a.elvers@rae.com

Abstract
The large volume of data generated during pipeline inspection is usually reduced to key indicators while valuable information remains hidden in intricate relations and trends between these data. Machine Learning (ML) is a branch of Artificial Intelligence (AI) that helps to discover these hidden patterns. ML is becoming a ubiquitous tool in data science. It provides meaningful information beyond traditional statistical analysis. In this paper, we present an application of supervised learning, a method where the algorithm learns from previous classifications, and then independently performs classifications in new datasets. First, we use a training dataset from a known database with flaws classified by operators and validated by supervisors. Then, we apply the trained algorithm in new datasets from a large inspection database. Our method shows an accuracy above 97% in both flaw type and acceptance criteria classifications.

Keywords: Machine learning, pipeline inspection, supervised learning, flaw type classification, acceptance criteria.

1. Introduction

The classification of indications in non-destructive testing (NDT) is a labor-intensive task, subject to operator bias. The advancements in the electronics and inspection techniques are ultimately bounded by a human interpretation step that can be improved by using machine learning algorithms.

Machine learning (ML) is a subfield of Artificial Intelligence (AI) which uses statistical techniques to give computer systems the ability to learn from data [1]. Two conventional methods in ML are supervised and unsupervised learning. In the former, we need the input data, and some output data to learn the correct answers (e.g., classification and regression) in the latter the algorithm learns from the input data without knowing the output data (e.g., cluster analysis and association).

Supervised learning simulates the NDT process where the operator labels the indications according to parameters like shape, location, and inspection method. Multiple variables are influencing this labeling: operator’s experience, inspection technique, material properties, etc. Supervised learning works on labeled data, which is the case of ultrasonic inspection by learning the patterns and associations between those labels. Once the algorithm has learned the classification model from a known dataset, it can be used in unclassified data. The known dataset is used to train the model and learn the classifications. In simple words, the algorithm learns from the indications classified by the operators, and then it could be applied to an unlabeled dataset and do the job with high accuracy.

In this paper with first describe the data preparation and the database system used by the UTScan® system, which allows the direct application of ML techniques to our current and past inspection databases. We then discuss the algorithm selection. The results section describes the accuracy in the classification with different datasets, and finally, we provide conclusions and future work.

2. Getting your data ready for Machine Learning

Data collected during the NDT inspection is crucial to ensure the safety of pipelines. However, the consistency of the results and the reliability of the findings are heavily dependent on the personnel performing the inspections and evaluation of the results [2]. The good news is that ML algorithms can operate over large datasets and learn the classification for multiple indications from different operators. Even, if some indications have been incorrectly classified, thanks to the central limit theorem, the error in large numbers will significantly reduce. Hence the classification accuracy will improve.

Pipeline inspection data is the perfect case to apply a supervised learning algorithm. The database holds information about the indications, such as depth, height, length, etc., together with the identification, e.g., incomplete fusion, porosity, lack of fusion, etc. These are the labels the operator used to classify the indications along with their evaluation as acceptable or rejectable.

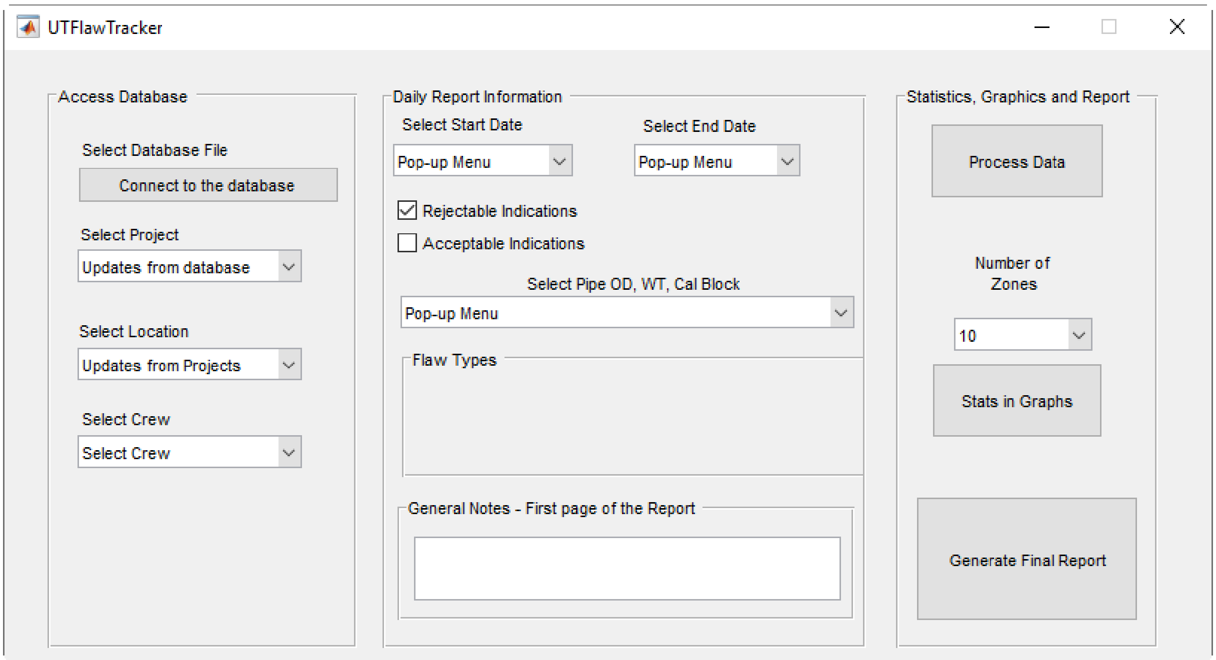

Figure 1: UT Flaw Tracker©, filters the dataset to feed the proper data into the ML algorithm.

In principle, ML learns from data, but you have to feed the right data into the algorithm to get valuable results. We have built an in-house statistical tool (UT Flaw Tracker©) which filters the data before the ML step. The UT Flaw Tracker© connects to the database and sequentially filters out the data by selecting: Project, Location, Crew, Dates, Evaluation (Rejectable, Acceptable), Pipe OD, Wall Thickness and Cal Block (you can have different pipe diameters in the same project, the same with WT and calibration blocks). After selecting these parameters, you are prompted to choose the flaw types or identified indications. A sample filtered looks like Table 1.

Table 1: A subset of the UT Flaw Tracker© output table used to perform statistical calculations and to feed the ML algorithm.

WeldNum	Result	Class	Ident	Label	WeldSide	Start	End	Length	Depth	Height	WallThickness	ScanDateTime
Weld01	Accepted	LB	IF	VOL-3	UpStream	274.0	282.0	8.0	4.3	1.0	19.1	9/28/2017 0:05
Weld02	Rejected	LB	IF	VOL-Cap	UpStream	366.0	475.0	109.0	1	1.0	19.1	9/28/2017 0:35
Weld03	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 0:38
Weld04	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 0:50
Weld05	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 1:34
Weld06	Rejected	LB	IF	TOFD	N/A	331.0	358.0	27.0	16.1	1.0	19.1	9/28/2017 1:54
Weld07	Accepted	LB	IF	TOFD	N/A	281.0	286.0	5.0	15.4	1.0	19.1	9/28/2017 2:07
Weld08	Accepted	LB	IF	TOFD	N/A	433.0	437.0	4.0	13	1.0	19.1	9/28/2017 2:24
Weld09	Rejected	LB	IF	TOFD	N/A	160.0	182.0	22.0	16.3	1.0	19.1	9/28/2017 2:45
Weld10	Rejected	VR	VOL	TOFD	N/A	238.0	250.0	12.0	19.1	3.5	19.1	9/28/2017 3:25
Weld11	Rejected	VI	P	TOFD	N/A	451.0	495.0	44.0	16.5	1.3	19.1	9/28/2017 3:37
Weld12	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 3:42
Weld13	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 4:18
Weld14	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 4:34
Weld15	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 4:55
Weld16	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 5:13
Weld17	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 5:27
Weld18	Accepted	LB	IF	VOL-Cap	UpStream	517.0	8.0	20.0	1.6	1.0	19.1	9/28/2017 5:55
Weld19	Accepted	LB	IF	TOFD	N/A	370.0	375.0	5.0	16	1.0	19.1	9/28/2017 6:12
Weld20	Accepted	LB	IF	VOL-1	UpStream	121.0	127.0	6.0	13.5	1.0	19.1	9/28/2017 6:27
Weld21	Rejected	LB	IF	TOFD	N/A	182.0	243.0	61.0	16.2	2.5	19.1	9/28/2017 6:46
Weld22	Accepted	LB	Geo	CL	UpStream	524.0	50.0	55.0	0	0.0	19.1	9/28/2017 7:02
Weld23	Accepted	LB	Geo	CL	DownStre	493.0	50.0	86.0	0	0.0	19.1	9/28/2017 7:28
Weld24	Accepted	LB	IF	VOL-4	UpStream	252.0	254.0	2.0	4.3	1.0	19.1	9/28/2017 7:42
Weld25	Accepted	LB	Geo	CL	DownStre	472.0	518.0	46.0	0	0.0	19.1	9/28/2017 7:54
Weld26	Rejected	VR	VOL	TOFD	N/A	258.0	270.0	12.0	19.1	3.5	19.1	9/28/2017 8:18
Weld27	Accepted	LB	Geo	CL	UpStream	426.0	451.0	25.0	0	0.0	19.1	9/28/2017 8:49
Weld28	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 9:11
Weld29	Accepted					0.0	0.0	0.0	0	0.0	19.1	9/28/2017 9:28
Weld30	Accepted	LB	Geo	CL	DownStre	483.0	62.0	108.0	0	0.0	19.1	9/28/2017 9:47
Weld31	Accepted	LB	Geo	CL	DownStre	509.0	57.0	77.0	0	0.0	19.1	9/28/2017 10:27
Weld32	Accepted	LB	Geo	CL	DownStre	445.0	33.0	117.0	0	0.0	19.1	9/28/2017 10:44
Weld33	Rejected	VR	VOL	TOFD	N/A	202.0	219.0	17.0	19.1	2.2	19.1	9/28/2017 11:13
Weld34	Accepted	LB	IF	Cap2	UpStream	250.0	279.0	29.0	2	1.2	19.1	9/28/2017 11:42
Weld35	Accepted	LB	Geo	CL	DownStre	467.0	520.0	53.0	0	0.0	19.1	9/28/2017 12:05
Weld36	Accepted	LB	Geo	CL	DownStre	476.0	14.0	67.0	0	0.0	19.1	9/28/2017 12:26
Weld37	Accepted	LB	Geo	VOL-4	DownStre	414.0	468.0	54.0	0	0.0	19.1	9/28/2017 12:43

The next steps involve the statistical analysis about the distribution of indications in the pipe, number of indications in the wall thickness and the frequency of occurrence of each flaw type and the percentage of the total amount. The statistical analysis already provides valuable information, but ML could be an additional tool for validation. Thus, we use the generated table to feed the ML algorithms.

3. Algorithm Selection

The algorithm selection is entirely data-driven. As there is no previous study on the dataset, the best option is to run different algorithms and compare their performances, to select the best-suited algorithm for your application — usually, the one consistently producing the best results.

We train the algorithm to learn from the inspection data to infer the classification of the indications with the same accuracy as the operators do or higher. The task can be described as:

I- Given the measured data from the UT system, classify the Flaw Types as Geometry, Incomplete Fusion, Porosity, etc. and evaluate the indication into Accepted it or Rejected.

The parameters we use to train the algorithm are called predictors; in this case:

- Flaw Identification
- Channel
- WeldSide
- Start
- End
- Length
- Depth
- Height
- WallThickness – Physical data

Another condition to impose to the algorithm is to be consistent and with an accuracy close to 100%; independently of your mood, the weather, your vision, or any other external situation affecting the inspection result and its reproducibility.

For the algorithm selection, we use the Classification Learner App in Matlab® [3] over a dataset of 766 indications.

We use the nine predictors listed above to train multiple algorithms, as is shown in Figure 2.

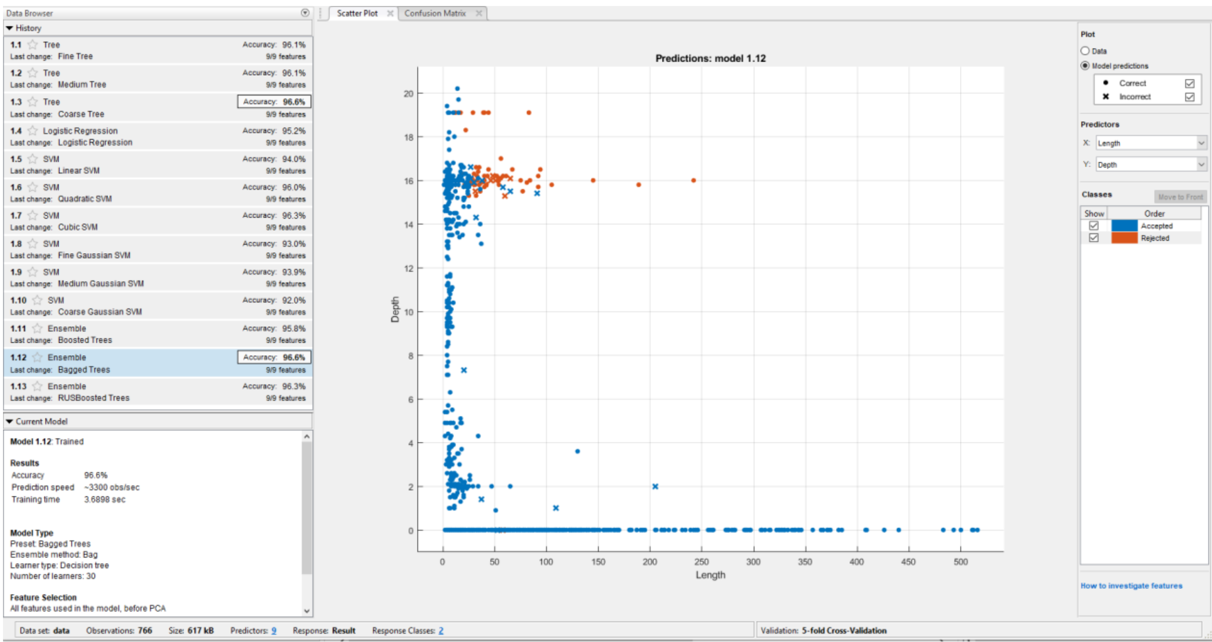

Figure 2: Training the multiple algorithms on the training dataset. An Ensemble method produces the highest accuracy of 96.6%.

The Confusion Matrix shows the true positive rate and false negative rate. We can interpret from the confusion matrix that for the accepted indications, the algorithm predicted class matches 99% of the true class, while only the 78% matches the rejected, meaning that we should expect more false negatives in rejected indications.

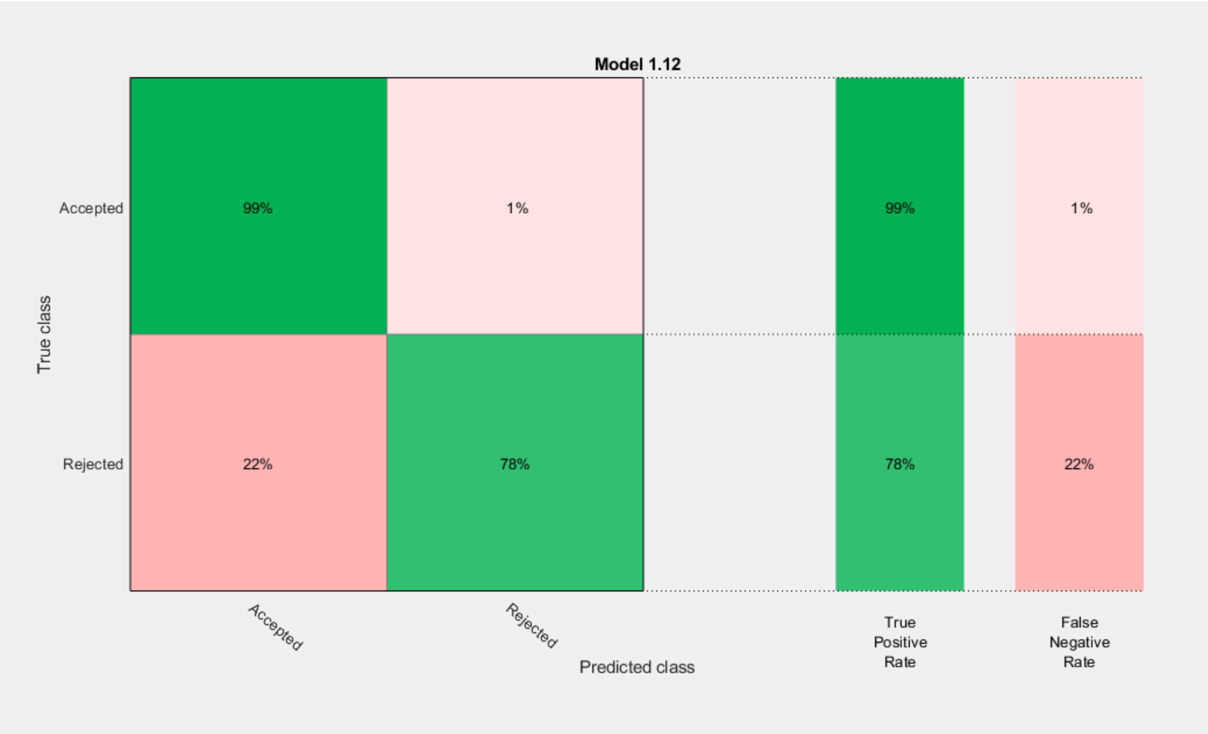

Figure 3. The Confusion Matrix.

The best performing algorithm was the ensemble Bagged Trees. With this algorithm, we create the model to run over the new unlabeled datasets.

4. Results

Once the Machine Learning algorithm produces a reasonable result, we process a new dataset with the Trained Algorithm.

A database from a recent project (2018) was used as a testing dataset. We ran the trained model over the 657 indications in this new dataset, hiding the labels and evaluation performed by the operators (flaw types and evaluation: accepted or rejected).

From 657 Indications, the ML algorithm correctly classified 654 in Accepted or Rejected for an accuracy of 99.5%. Only three indications were accepted that the algorithm thinks we should reject. After consultation with the specialist, these were close-calls. These indications were correctly classified by the operators but were so close to the acceptable limit that another operator could have had a different opinion.

For the flaw type, the accuracy was 97.56%, the algorithm misclassified 16 indications, but after comparing the results in Table 2, many of the indications were labeled as non-classified

(NC) by the ML algorithm. This NC label comes from the training dataset where operators marked but did not label the indication. In practice, only two indications (Geo as IF and IF as Non-Classified), were wrongly classified by the algorithm leading to an accuracy of 99.69%.

Table 2: Results of the ML Algorithm. Misclassified labels, 16 out of 657 indications. See the explanation on the text for the NC label.

Operator	ML Algorithm
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'IF' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC'}
{'Geo'}	{'NC' }
{'Geo'}	{'NC'}
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'Geo'}	{'NC' }
{'IF' }	{'Geo' }

3. Conclusions

The machine learning algorithm classified the new dataset with high accuracy, yet, at this stage, we advise against completely unsupervised classifications, because the method is intended as a guide instead of a fully automated blind approach.
We see the ML algorithm as a tool for the operator in the classification of flaw types and evaluation. Also, the algorithm could be used to validate the acceptance criteria.
We are working to integrate the Machine Learning functionality into the current version of the UTScan® system as a validating method.

References

[1] A. Ng, *Machine Learning Yearning*. Online Draft. [book.ml], 2017.

[2] A. Gonzalez Rodriguez, A. Gosselin, R. Rheaume, and N. Harrap, “Computational intelligence approaches for data analysis, the next step of innovation for advanced UT techniques in NDT,” in *ECNDT 2018: 12th European Conference on Non-Destructive Testing (ECNDT 2018), Gothenburg 2018, June 11-15*, 2018, pp. 1–2.

[3] MATLAB, *version 9.5.0.1033004 (R2018b)*. Natick, Massachusetts: The MathWorks Inc., 2018.