Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2022) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)

Predicting Yarn Breaks in Textile Fabrics: A Machine Learning Approach

João Azevedo[a], Rui Ribeiro[a,b], Luís Miguel Matos[b], Rui Sousa[c], João Paulo Silva[d], André Pilastri[a], Paulo Cortez[b,*]

[a]*EPMQ - IT Engineering Maturity and Quality Lab, CCG ZGDV Institute, Guimarães, Portugal*
[b]*ALGORITMI Center, Dep. Information Systems, University of Minho, Guimarães, Portugal*
[c]*Somelos, Ronfe, Portugal*
[d]*Fluxodata, Ronfe, Portugal*

Abstract

In this paper, we propose a Machine Learning (ML) approach to predict faults that may occur during the production of fabrics and that often cause production downtime delays. We worked with a textile company that produces fabrics under the Industry 4.0 concept. In particular, we deal with a client customization requisite that impacts on production planning and scheduling, where there is a crucial need of limiting machine stoppage. Thus, the prediction of machine stops enables the manufacturer to react to such situation. If a specific loom is expected to have more breaks, several measures can be taken: slower loom speed, special attention by the operator, change in the used yarn, stronger sizing recipe, etc. The goal is to model three regression tasks related with the number of weft breaks, warp breaks, and yarn bursts. To reduce the modeling effort, we adopt several Automated Machine Learning (AutoML) tools (H2O, AutoGluon, AutoKeras), allowing us to compare distinct ML approaches: using a single (one model per task) and Multi-Target Regression (MTR); and using the direct output target or a logarithm transformed one. Several experiments were held by considering Internet of Things (IoT) historical data from a Portuguese textile company. Overall, the best results for the three tasks were obtained by the single-target approach with the H2O tool using logarithm transformed data, achieving an R^2 of 0.73 for weft breaks. Furthermore, a Sensitivity Analysis eXplainable Artificial Intelligence (SA XAI) approach was executed over the selected H2OAutoML model, showing its potential value to extract useful explanatory knowledge for the analyzed textile domain.

© 2022 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the KES International.

Keywords: Machine Downtime; Yarn Breaks; Explainable Artificial Intelligence; Regression; Automated Machine Learning.

* Corresponding author. Tel.: +351 253 510 309 ; fax: +351 253 510 300
E-mail address: pcortez@dsi.uminho.pt

1877-0509 © 2022 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the KES International.

1. Introduction

Nowadays, there is a pressure in the manufacturing industry to increase efficiency and reduce lead times. Within this context, production downtime is one of the most important issues in manufacturing because it directly affects productivity, efficiency, and profitability. It can be caused by planned maintenance, tool breaks, adjustments and several human factors. Machine downtime is one of the attributable causes of variation in a manufacturing system, resulting in poor reliability of the production schedule [21]. Thus, reducing downtime will increase machine availability which in turn increases throughput reducing order lead times and increasing customer satisfaction [22].

In this work, we collaborate in the digital transformation process of Somelos, a Portuguese textile company that produces fabrics, in order to improve the autonomy and performance levels of production planning and control. We particularly focus on an issue raised by the organization and that is related to machine downtime regarding some specific operating methods. The loom operating method consists in inserting a weft thread into a warp. For this process to be possible, the healds cause some of the warp yarns to rise, separating the warp into two sheets; the weft is inserted, passing the thread through the shed along with the fabric; the reed makes a beat, pushing the inserted thread against the already formed fabric. This process can generate breaks in the warp yarns. The tensions to which the yarn is subjected in this process will highlight the weak points of the raw material. Thus, a yarn with thin places will tend to break when subjected to tension that would not be a problem for other areas of the yarn. Increasing the speed of the process will increase the friction between the threads (in the process of shedding). Therefore, it will increase the tensions of the process, leading to more breaks. The thick places and neps tend to increase friction between threads when producing a fabric with high thread density [2]. This process can cause three major problems related to yarns: weft breaks, warp breaks and yarn burst. When these faults occur, the operator must stop the machine in order to reattach the broken yarns and only then she/he can resume the production. Even if this time is included in the article expected efficiency, the ability to predict a more accurate efficiency will produce a better planning output.

The company has already implemented data collectors in the looms. This information is online and has given the ability to react to several production events. But all of them are related to past events. By using data collected by Internet of Things (IoT) sensors from the textile machines and Machine Learning (ML) algorithms, the goal of this research is to predict the occurrence of the three types of fabric faults, allowing to better support production planning and also identify the main factors that influence production faults. The collected data is related to fabrics characteristics and the loom specifications extracted from the company Enterprise Resource Planning (ERP). To reduce the ML modeling effort, we adopt three Automated ML (AutoML) tools, namely H2O, AutoGluon and AutoKeras, which are complemented by a manually designed Deep FeedForward Neural Network (DFFN). Furthermore, the best predictive model is further analyzed by using a Sensitivity Analysis eXplainable Artificial Intelligence (SA XAI) method [8], which allows to measure the overall impact of the selected inputs on the predictions. This paper is organized as follows: Section 2 introduces the related work; Section 3 describes the data related to fabrics and production machines, the ML approaches and the evaluation; Section 4 details the obtained results; and finally Section 5 presents the main conclusions.

2. Related Work

There are several characteristics of a yarn, such as uniformity and hairiness, that can increase tension in weaving, and that combined with the winding tension of the beam can cause faults that stop production, such as warp and weft breaks. In this regard, several authors presented empirical, statistical and instrumental methodologies to solve the warp breaking problem [9]. Over the last years, different approaches based on ML techniques were also proposed to solve this challenge. For instance, [5] proposed a program based on mathematical models through relationships between specific elements of the yarn and its breaking strength, with the goal of predicting yarn breaks in the weaving process, and concluded that weft breaks are significantly influenced by the weft insertion length and type of insertion, given that increasing the machine width will have a higher probability of breakage. In [26] the authors employed a feedforward back-propagation Neural Network (NN) to forecast the rate of warp breakage, obtaining a model that used a single sigmoid hidden layer with four neurons, allowing to infer that the prediction of warp breakage is feasible (a correlation coefficient of 99.5% was obtained between the predicted and true values). In [20] the authors also adopted an NN to predict yarn break elongation and then inspected the influence of the input parameters and NN properties on prediction. On the neural functions used, the best performance was achieved with Purelin algorithm and the best NN architecture was obtained using Levenberg-Marquardt training function. The six most influential factors of the obtained NN were: yarn twist, yarn count, fiber elongation, length, length uniformity and spindle speed. In another study, [27] proposed the use of NN with feedforward back-propagation to predict the warp breakage rate using the yarn quality index as input. The selected ML model contained only one hidden layer and eight neurons, obtaining a correlation coefficient of R^2=99.3%. These works support the idea that NNs can forecast warp breaking rates and that they are a useful approach with great potential for the textile industry. The ANOVA technique was used in [1] work to detect the effects of the variables on yarn breaks on the warping machine and a regression model to predict the number of yarn breaks, with the goal of inferring the impact of cotton/polyester blending ratio, cotton type, yarn twist, and yarn count on yarn breaks on the warping machine. In terms of the impact factor, the findings reveal that the cotton/polyester blending ratio has a considerable impact on yarn breaks, and that the number of yarn breaks varies directly with single and plied yarn counts. According to [10], the most important factors affecting the efficiency of an air-jet weaving machine are warp and weft breaks. The authors used a back-propagation algorithm to train a NN to predict the number of weft breaks per million meters, and obtained a R^2 of 0.955 between the actual number of weft breaks and the predicted value, indicating that weft breaks can be predicted using NNs. In another study, [23, 24] proposed the use of automatic methods to predict different properties of woven fabrics based on design and finishing features using an Automated Machine Learning (AutoML) during the modeling stage of the CRISP-DM.

More recently, [25] developed a linear regression model to check if there is a link between warp breakage on the weaving machine and mechanical yarn stretch. This link was corroborated by the obtained results (R^2= 84.4%).

Within our knowledge, none of the related works was conducted under an Industry 4.0 environment, which allows to easily collect big data. Moreover, the related works did not employ current state-of-the-art ML approaches, such as AutoML and DL, which is addressed in this work.

3. Material and Methods

3.1. Textile and Machine Production Data

For the creation of the analyzed dataset, we have merged data attributes concerning the characteristics of the fabrics and the textiles machines. Two tables from the textile company database were utilized to collect fabric characteristics: one with 20,999 records that showed generic characteristics of all fabrics and another with 28,249 records that corresponded to the features of the fabrics produced on the machines. On the other hand, the data related to the machine characteristics came from four different database tables: one with general machine characteristics (264 records), two tables with the link between work orders and machines (24,484 and 30,964 records) and one with two years of sensory data on machine downtime, with a total of 5,517,384 records corresponding to 29,959 work orders, of which only 11,855 correspond to the remaining tables.

For extraction purposes, a combination of inner joins operations were used between the primary keys of the tables to create just one data set, resulting in 1,966,217 records. To get the number of weft breaks, warp breaks, and yarn bursts, the data had to be grouped using code_fabric, idShift, the work order identifier, and the MachineTypeCode, which resulted in three new columns representing the number of warp breaks, weft breaks and yarn bursts, reducing the dataset to 1,502 records.

Apart from the three targets (number of warp breaks, number of weft breaks and number of yarn bursts), the final dataset has twenty different features, 18 of which are numeric, one categorical, and one nominal. In particular, the work order identifier feature was discarded because it consists of a simple incremental identification code. Table 1 summarizes the attributes of the final dataset. The first group of attributes are related to fabric characteristics and the second group is related to textile machine characteristics. The value ranges for the three target attributes (the number of Warp Breaks, Weft Breaks and Yarn Bursts) are: [1, 226], [1,236], and [1, 15].

3.2. Data Preprocessing

The data preprocessing phase was divided into two approaches which are factor of comparison in the Table 2. In approach **A**, the data preprocessing phase involved the transformation of the categorical and nominal data as well as the

Table 1: List of input attributes used for the regression tasks.

	Attribute	Description (data type)	Range
Fabric Characteristics	code_fabric	Fabric type code (numeric)	[4009531, 5152412]
	fabrictype	Code that identifies if the fabric is raw or finished (nominal code)	2 Levels
	WarpDrawingCode	Code that identifies the warp design (categorical code)	508 Levels
	WeftDrawingCode	Code that identifies the weft design (numeric)	[1, 999]
	WidthFabric	Fabric width in centimeters (numeric)	[127, 187]
	TotalWeft_Yarns	Quantity of weft yarns (numeric)	[2992, 19000]
	Nweft_Yarn_Inch	Number of weft yarns per inch (numeric)	[51, 318]
	TotalWarp_Yarns_Rapport	Quantity of warp yarns (numeric)	[1, 1880]
	Reed Width Fabric	Width of the reed for the fabric in centimeters (numeric)	[155.59, 235]
	N_EndHeald	Quantity of healds in the extremity (numeric)	[0, 4]
	Linear_Weight_CrudeItem	Linear weight of the raw material (numeric)	[105, 534]
	N_Heald_FabricBody	Quantity of healds in the body of the fabric (numeric)	[4, 28]
	Perc_QualMaterialWeaving	Percentage of second quality article in the weaving mill (numeric)	[0.25, 4]
Machine Characteristics	idShift	Identification working shift (numeric)	[1, 3]
	MachineTypeCode	Code that identifies the type of loom (numeric)	[21, 81]
	Npicks_Inch_RawMat	Number of picks per inch for raw material (numeric)	[34, 146{]
	Npicks_Inch_Warp_FinishedMat	Number of warp picks per inch for finished material (numeric)	[37, 150]
	Weight_Finished_Fabric	Weight of finished article (numeric)	[64, 336]
	MachineSpeed	Speed of the loom (numeric)	[400, 900]
	Reed Width	Width of the reed in centimeters (numeric)	[190, 260]

numerical attributes that represented identification codes such as code_fabric, MachineTypeCode and WeftDrawingCode. For the transformation of the nominal attribute, taking into account that it had only two levels, a label-encoder was used. In the remaining transformations, the Inverse Document Frequency (IDF) proposed by [4] was used. IDF is a data transformation process where a categorical value is encoded according to the following equation:

$$IDF(l) = ln(\frac{N}{f_l}) \tag{1}$$

where N denotes the total number of instances (values), and f_l represents the number of occurrences of level l in the training data. When using this transform, a closer numeric value to 0 means that the level is persistent in the data. The higher the value, the less frequent the level is, with the less frequent levels being grouped closer. This transformation methods is particularly useful when the cardinality of categorical features is substantially high [16, 15]. In order to standardize the values of the data set, a standard scaler was applied to all attributes (except targets). Approach **B** followed all the previous procedures, but the function $\log(y + 1)$ was applied to the output targets. This transformation is commonly adopted when the target distribution is skewed to the left, as shown in Fig. 1. It should be noted that once a ML is trained, the inverse logarithm function is applied to the model predictions, in order to analyze the approach **B** results using the original target scale (as performed for approach **A**). The application of the logarithmic function made the distribution more centered, removing the left skewed distribution that was mainly seen in the number of warp breaks and the number of weft breaks, as presented in Fig. 2.

3.3. Machine Learning Methods

Considering that the same fabric can have all three faults in a single production order, this research compares two main ML approaches: single-target and multi-target. The former assumes a distinct model selection and fit for each of the three analysed tasks, while the latter, known as Multi-Target Regression (MTR), adopts a single ML model with three outputs, one for each task. The implementation of the AutoML procedure automatically selects the best among several state-of-the-art ML algorithms and allowed us to better focus on feature engineering, which is a non-trivial task in this domain.

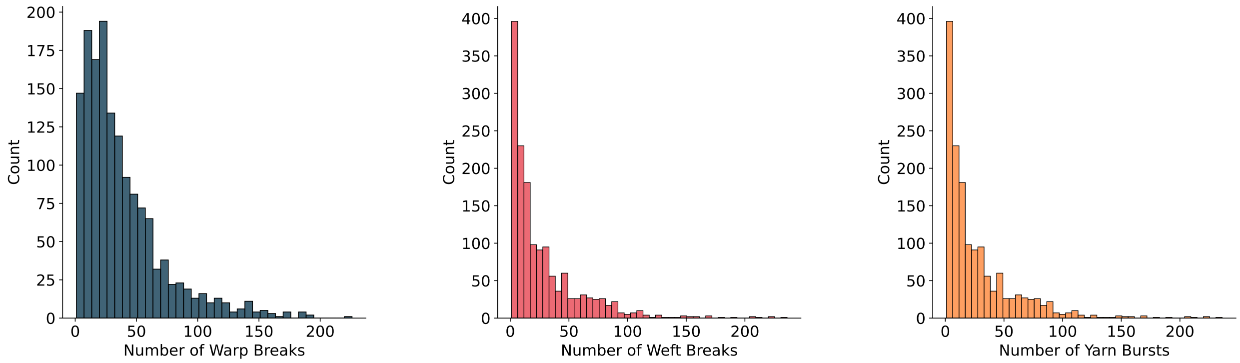

Fig. 1: Distribution of the warp break (left), weft break (middle) and yarn burst (right) regression targets

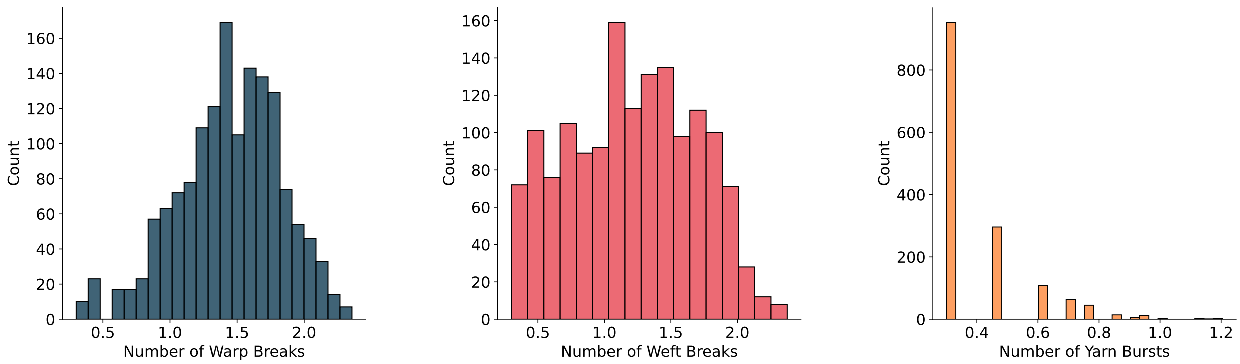

Fig. 2: Distribution of warp break (left), weft break (middle) and yarn burst (right), after the application of the logarithmic function.

The single-target experiments were run through two AutoML tools, namely AutoGluon [11] and H2O AutoML [6], where a 10-fold internal validation was employed to select and tune the best ML model. The AutoML tools were set up to automatically select the optimal model and its hyperparameters for each fold by minimizing the Mean Absolute Error (MAE) measure, with a maximum execution duration of 1 hour per fold. In each iteration of the H2O AutoML execution, the following ML families were accessible during the search: Distributed Random Forest (DRF), Generalized Linear Model with Regularization (GLM), XGBoost, Gradient Boosted Machines (GBM) and Stacked Ensembles (SE). The SE can be build by using all searched ML models (all) or by using only the best model for each ML algorithm type (best of family). In turn, the ML algorithms searched by the AutoGluon tool were: LightGBM, CatBoost Boosted Trees, RF, Extra Trees (XT), k-NN, Multiple Linear Regression (MR), as well as a DL dense architecture that employs ReLU activation functions, dropout regularization, and batch normalization layers, and also heuristics to adjust the hidden layer sizes. These AutoML tools provide a performance based stopping criterion, which will stop the search process when the performance does not improve by a specified amount. In our case, we assumed the default baseline performance criterion for each AutoML tool.

The AutoKeras tool [14] and a deep Multilayer Perceptron (MLP), also known as Deep FeedForward Neural Network (DFFN) [13], were adopted for the MTR approach that handled all three targets (described in Section 3.1) by using the same ML model.

The AutoKeras is an AutoML tool for DL, thus it automatically searches for the best DFFN model, tuning the number of dense layers, units, type of activation functions utilized, dropout levels, and other DL hyperparameters through a Bayesian Optimization [12]. The AutoKeras tool was configured similarly to the single-target AutoML tools, namely the search used a 10-fold internal validation and the selection criterion was the average (of the three tasks) MAE value. By default, the AutoKeras model tests a maximum of 100 models and chooses the model that produces the best loss value. At the end and for each of the ten folds, the best models were stored on the machine.

The manually designed DFFN adopted the Keras Python module and consisted of a setup that is identical to that used in [17, 18, 19], with the exception that the DFNN contains three output nodes with the ReLU activation function. In this work, the manually designed DFFN uses a triangular shape MLP, in which each subsequent layer size is smaller: $Input > L_1 > ... > L_H > 3$. This structure is composed by $H = 8$ hidden layers being composed as follows: $(I, 1024, 512, 256, 128, 64, 32, 16, 8, 3)$.

3.4. Evaluation

In order to obtain a robust evaluation, all regression approaches were evaluated by using a 10-fold external cross-validation method. To measure the quality of the results, the Mean Absolute Error (MAE), the Normalized MAE (NMAE), and the R^2 for each fold were calculated and then averaged over the 10 iteration folds. The NMAE measure normalizes the MAE by the output target range on the test set, resulting in a scale-independent percentage that is easy to understand and is represented by the following formula:

$$NMAE = \frac{MAE}{(y_{max} - y_{min})} \tag{2}$$

where the y_{max} and the y_{min} represents the highest and the lowest values of the target, respectively. Note that better predictions are indicated by higher R^2 values (the perfect value is 1), as well as by lower NMAE and MAE values (the perfect value is 0).

4. Results

Table 2 summarizes the obtained predictive results. For each predicted task(**Objective**), we detail the used ML technique (**Tools**) and the two strategies (**Str**) used. As previously explained, the results are shown in terms of the average of the test scores for the external 10 folds (**MAE**, **NMAE** and $\mathbf{R}^2$). For the MAE measure, the table also includes the standard deviation of the 10-fold external iterations ($\pm s$, where s is the standard deviation value) and the mode of the most selected model for each approach. The best values are highlighted by using a **boldface** text font.

When analyzing the obtained results, it becomes clear that the single-target approach presents better predictive performances when compared with the multi-target approach, independently of the preprocessing strategy of AutoML tool used. Overall, and considering both MAE based and R^2 measures, the best ML approach was obtained by the single-target H2OAutoML tool and output logarithm transform (strategy **B**). Indeed, this approach obtained: the best NMAE and R^2 results for the Weft Breaks task, the best NMAE for the Warp Breaks target, with the R^2 being only 1 percentage point worst than the best value; the third best NMAE result for the Yarn Bursts task, while obtaining an R^2 value of 0.43 and that is substantially higher than the one obtained for the methods with better NMAE values.

For demonstration purposes, Fig. 3 shows the Regression Error Characteristic (REC) curves [3] for the three target prediction and 7^{th} external k-fold iteration. Each REC curve plots the percentage of correctly predicted examples (y-axis) for a given absolute error tolerance (x-axis). For instance, the left plot of Fig. 3 reveals that more than 70% of the warp break predictions are correct when adopting a small tolerance of 0.4 points. To complement the visualization of the obtained results, Fig. 4 presents the scatter plots of the measured (x-axis) versus the predicted values (y-axis). Visually, it can be seen that the predictions for the warp and weft breaks are close to the ideal prediction diagonal line. While the same effect is not that visible for the yarn burst predictions, it should be noted that in the left of Fig. 4 several of the predicted points do overlap. Thus, a better visual analysis is obtained in the right of Fig. 3, showing that interesting predictions were obtained (e.g., more than 80% of the burst predictions are correct for a 0.4 tolerance). Moreover, we highlight that the obtained predictive results were shown to the textile production experts, which provide a positive feedback, found them very interesting to support production planning.

To further demonstrate the value of the selected predicted models (H2O tool and logarithm transform), we applied a SA XAI approach to the 7^{th} external fold iteration results, namely the 1-D SA, as implemented by the `rminer package` [7, 8]. For this iteration, the AutoML tool selected a GBM model. The left of Fig. 5 plots the relevance of the input variables (total of 20 input features). For instance, the most influential input is related with the number of yarns on the weft by inches (total relevance of 14%). Moreover, the top 9 inputs account for 66% of the total influence in the GBM model. As for the right of Fig. 5, it shows the Variable Effect Characteristic (VEC) curves for the top 5 relevant input variables. The VEC curve shows the average influence of one input in the output target when varying its

Table 2: Obtained predictive results (best results per task are in **bold**)

		Regression Metrics				
Objective	**Tools**	**Str**	**MAE**	**NMAE**	**R^2**	**Most Selected Model**
WarpBreaks	H2O	A	14.11 ± 0.77	8.18%	**0.58**	GBM
		B	**13.93** ± 1.02	**8.07%**	0.57	GBM
	AutoGluon	A	14.30 ± 0.76	8.28%	**0.58**	SE (all)
		B	14.17 ± 0.93	8.22%	**0.58**	SE (all)
	Keras	A	17.82 ± 1.55	10.19%	0.41	DFFN
		B	18.51 ± 1.58	10.58%	0.39	DFFN
	AutoKeras	A	17.97 ± 2.23	10.33%	0.40	DFFN
		B	19.20 ± 2.09	10.97%	0.35	DFFN
WeftBreaks	H2O	A	10.96 ± 0.81	6.59%	0.72	GBM
		B	**10.65** ± 0.98	**6.47%**	**0.73**	SE (best of family)
	AutoGluon	A	11.25 ± 1.02	6.79%	0.71	SE (all)
		B	10.77 ± 1.13	6.50%	0.72	SE (all)
	Keras	A	15.05 ± 2.58	8.99%	0.44	DFFN
		B	14.69 ± 2.35	8.64%	0.46	DFFN
	AutoKeras	A	13.27 ± 1.33	8.03%	0.60	DFFN
		B	15.29 ± 1.23	9.09%	0.45	DFFN
YarnBurst	H2O	A	0.69 ± 0.08	7.93%	**0.44**	GBM
		B	**0.68** ± 0.09	7.79%	0.43	XGBoost
	AutoGluon	A	0.68 ± 0.12	**7.68%**	0.19	SE (all)
		B	0.68 ± 0.08	7.81%	0.32	SE (all)
	Keras	A	0.83 ± 0.12	8.74%	0.10	DFFN
		B	0.76 ± 0.11	7.70%	0.17	DFFN
	AutoKeras	A	0.84 ± 0.12	8.55%	0.23	DFFN
		B	0.73 ± 0.06	**7.52%**	0.27	DFFN

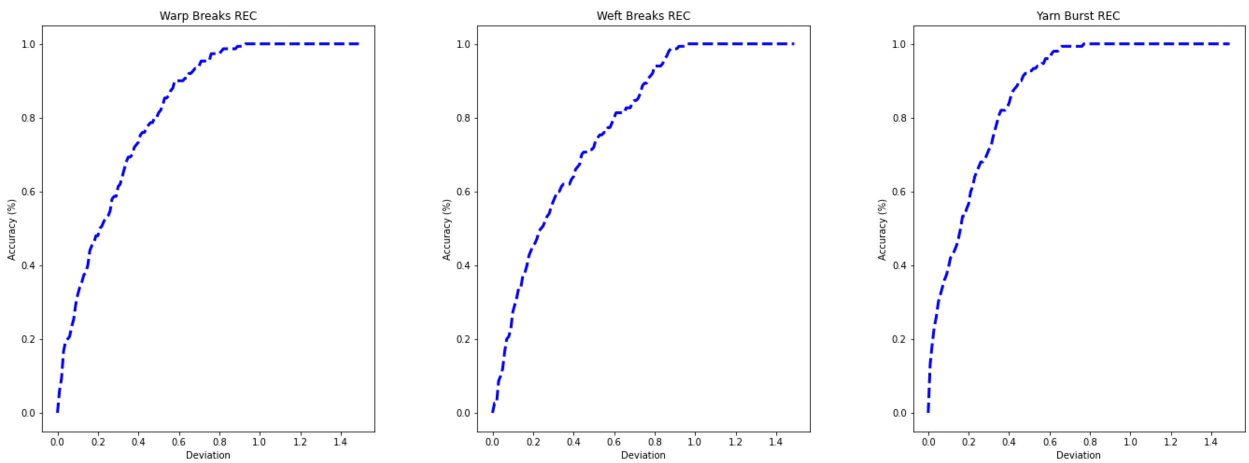

Fig. 3: REC curves for the warp break (left), weft break (middle) and yarn burst (right) predictions.

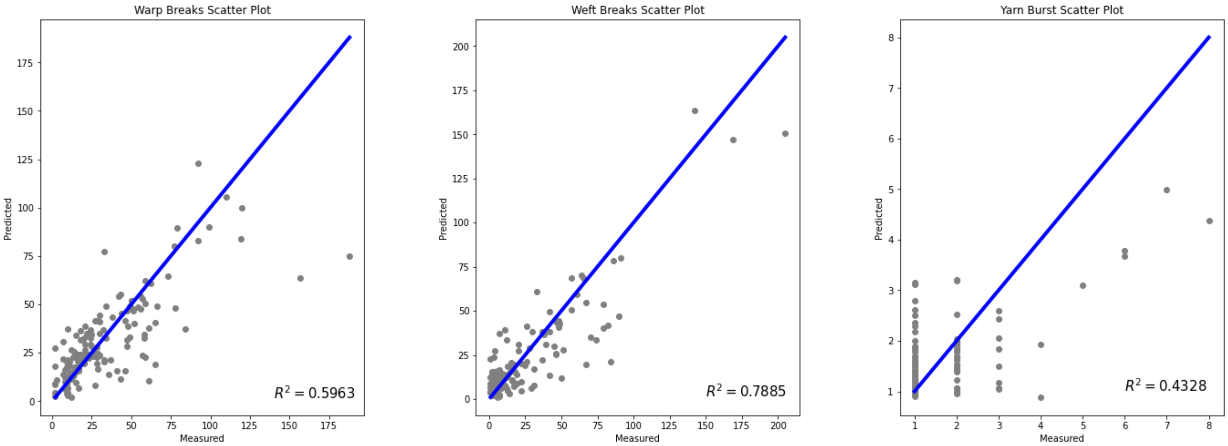

Fig. 4: Regression scatter plot of best models for the warp break (left), weft break (middle) and yarn burst (right) predictions.

range through $L = 7$ distinct levels. The plot clearly reveals that the most influential input (Nweft Yarn Inch) produces the largest GBM output response change (thus impacting more on the model). In general, an increase in the numeric input also produces an increase in terms of the number of weft breaks. The obtained XAI knowledge was also provided to the textile production experts, which confirmed that both input influence and input effects are in accordance with their expert empirical knowledge.

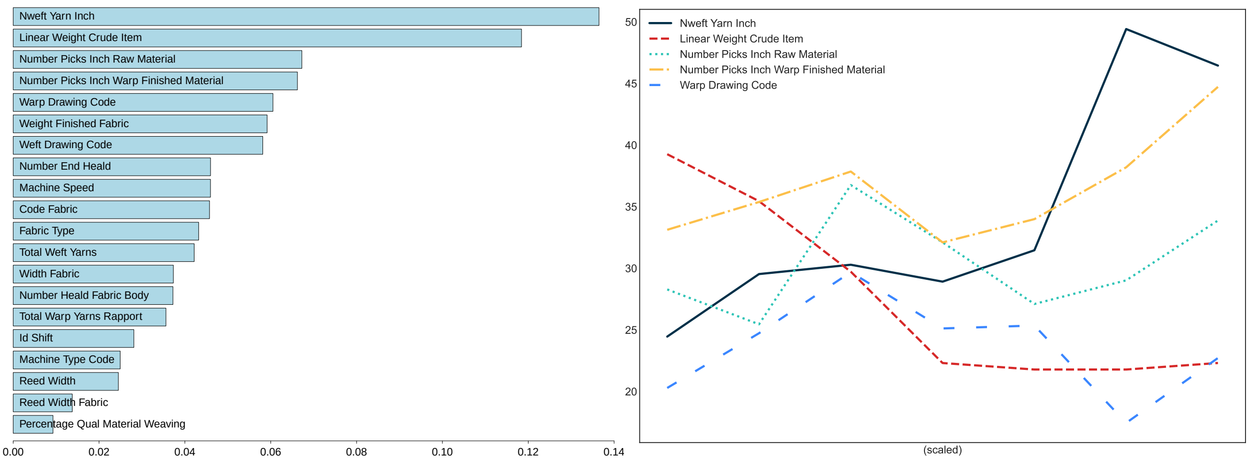

Fig. 5: Extracted XAI knowledge from the GBM model using the SA XAI method: input variable relevance (left) and top 5 VEC curves (right).

5. Conclusions

Under the Industry 4.0 paradigm, there is currently an opportunity to automatically collect data using IoT sensors linked to production machines and then employ ML approaches to increase efficiency and reduce lead times. In this work, we collaborate in the digital transformation process of Somelos, a Portuguese textile company that produces fabrics, in order to improve the autonomy and performance levels of their production planning and control. In particular, we focus on three regression tasks that can improve the planning of fabric production: the number of warp breaks, weft breaks and yarn bursts. Using real data, collected from IoT devices and the analyzed production system database, we explore two preprocessing strategies (with and without a target logarithm transform) and two main ML fitting approaches (single and multi-target). In order to reduce the ML modeling effort, we compared several AutoML tools, namely H2O, AutoGluon and AutoKeras. Overall, the best predictive results were obtained by the single-target approach, assuming the H2O tool and a logarithm target transform. Interesting performances were achieved, resulting in a NMAE value that ranges from 6.5% to 8.0% and R^2 that varies from 0.43 to 0.73. The selected model was further analyzed by using a SA XAI procedure, allowing to demonstrate valuable knowledge that can be extracted from the fitted ML. For instance, the increase in the number of weft yarns per inch tends to increase the number of warp breaks.

The obtained ML results were discussed with the textile manufacturer experts, which returned a positive feedback. Indeed, in future work, we aim to deploy the proposed ML approach in the real textile industrial setting.

Acknowledgments

This work is supported by the European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) [Project PPC4.0 - Production Planning Control 4.0; Funding Reference: POCI-01-0247-FEDER-069803].

References

[1] Almetwally, A.A., Mourad, M., Mohammed, A.E.E., 2013. A study of yarn breaks on warping machines. Life Science Journal 10, 108–114. URL: http://www.lifesciencesite.com/lsj/life1001/016_12676life1001_108_114.pdf.

[2] Araújo, M.d., Rocha, A.M., 1989. Tecnologia da tecelagem. DGI, Lisboa. URL: http://catalogo.biblioteca.min-economia.pt/cgi-bin/koha/opac-detail.pl?biblionumber=137876. vol. 1. - 257, [1] p.

[3] Bi, J., Bennett, K., 2003. Regression error characteristic curves, pp. 43–50. URL: https://www.aaai.org/Papers/ICML/2003/ICML03-009.pdf.

[4] Campos, G.O., Zimek, A., Sander, J., Campello, R.J.G.B., Micenková, B., Schubert, E., Assent, I., Houle, M.E., 2016. On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. Data Mining and Knowledge Discovery 30, 891–927. URL: https://doi.org/10.1007/s10618-015-0444-8.

[5] Cioară, I., Cioară, L., Onofrei, E., 2004. Forecast of Yarn Breakages During The Weaving Process. Research Journal of Textile and Apparel 8, 20–24. doi:10.1108/RJTA-08-01-2004-B003.

[6] Cook, D., 2016. Practical machine learning with H2O: powerful, scalable techniques for deep learning and AI. O'Reilly Media. URL: https://books.google.pt/books?id=C5amDQAAQBAJ.

[7] Cortez, P., 2020. rminer: Data Mining Classification and Regression Methods. URL: https://CRAN.R-project.org/package=rminer. r package version 1.4.6.

[8] Cortez, P., Embrechts, M.J., 2013. Using sensitivity analysis and visualization techniques to open black box data mining models. Inf. Sci. 225, 1–17. doi:10.1016/j.ins.2012.10.039.

[9] Das, B., Ishtiaque, S., Rengasamy, R., Hati, S., 2011. Prediction of Weavability of Warp Yarns: A Critical Review. Research Journal of Textile and Apparel 15, 38–49. doi:10.1108/RJTA-15-03-2011-B005.

[10] El-Tarfawy, S.Y., 2016. Prediction of Weft Breaks in Air Jet Weaving Machine by Artificial Neural Network. Journal of American Science doi:10.7537/marsjas12051605.

[11] Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., Smola, A., 2020. AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data URL: http://arxiv.org/abs/2003.06505.

[12] Ferreira, L., Pilastri, A., Martins, C.M., Pires, P.M., Cortez, P., 2021. A comparison of automl tools for machine learning, deep learning and xgboost, IEEE. pp. 1–8. doi:10.1109/IJCNN52387.2021.9534091.

[13] Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press. URL: http://www.deeplearningbook.org.

[14] Jin, H., Song, Q., Hu, X., 2019. Auto-keras: An efficient neural architecture search system, ACM. pp. 1946–1956. doi:10.1145/3292500.3330648.

[15] Matos, L.M., Cortez, P., Mendes, R., 2020. Cane - Categorical Attribute traNsformation Environment. URL: https://pypi.org/project/cane/.

[16] Matos, L.M., Cortez, P., Mendes, R., Moreau, A., 2018. A comparison of data-driven approaches for mobile marketing user conversion prediction, in: Jardim-Gonçalves, R., Mendonça, J.P., Jotsov, V., Marques, M., Martins, J., Bierwolf, R.E. (Eds.), 9th IEEE International Conference on Intelligent Systems, IS 2018, Funchal, Madeira, Portugal, September 25-27, 2018, IEEE. pp. 140–146. doi:10.1109/IS.2018.8710472.

[17] Matos, L.M., Cortez, P., Mendes, R., Moreau, A., 2019a. Using deep learning for mobile marketing user conversion prediction, in: International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019, IEEE. pp. 1–8. doi:10.1109/IJCNN.2019.8851888.

[18] Matos, L.M., Cortez, P., Mendes, R.C., Moreau, A., 2019b. Using deep learning for ordinal classification of mobile marketing user conversion, in: Yin, H., Camacho, D., Tiño, P., Tallón-Ballesteros, A.J., Menezes, R., Allmendinger, R. (Eds.), Intelligent Data Engineering and Automated Learning - IDEAL 2019 - 20th International Conference, Manchester, UK, November 14-16, 2019, Proceedings, Part I, Springer. pp. 60–67. doi:10.1007/978-3-030-33607-3_7.

[19] Matos, L.M., Domingues, A., Moreira, G., Cortez, P., Pilastri, A.L., 2021. A comparison of machine learning approaches for predicting in-car display production quality, in: Intelligent Data Engineering and Automated Learning - IDEAL 2021 - 22nd International Conference, IDEAL 2021, Manchester, UK, November 25-27, 2021, Proceedings, Springer. pp. 3–11. doi:10.1007/978-3-030-91608-4_1.

[20] Mwasiagi, J.I., Huang, X., Wang, X., 2008. Performance of neural network algorithms during the prediction of yarn breaking elongation. Fibers and Polymers 9, 80–86. doi:10.1007/s12221-008-0013-5.

[21] Nwanya, S., Achebe, C., Ajayi, O., Mgbemene, C., 2016. Process variability analysis in make-to-order production systems. Cogent Engineering 3, 1269382. doi:10.1080/23311916.2016.1269382.

[22] Nwanya, S., Udofia, J., Ajayi, O., 2017. Optimization of machine downtime in the plastic manufacturing. Cogent Engineering 4, 1335444. doi:10.1080/23311916.2017.1335444.

[23] Ribeiro., R., Pilastri., A., Moura., C., Rodrigues., F., Rocha., R., Cortez., P., 2020. Predicting the tear strength of woven fabrics via automated machine learning: An application of the crisp-dm methodology, in: Proceedings of the 22nd International Conference on Enterprise Information Systems - Volume 1: ICEIS,, INSTICC. SciTePress. pp. 548–555. doi:10.5220/0009411205480555.

[24] Ribeiro, R., Pilastri, A., Moura, C., Rodrigues, F., Rocha, R., Morgado, J., Cortez, P., 2020. Predicting physical properties of woven fabrics via automated machine learning and textile design and finishing features, in: Maglogiannis, I., Iliadis, L., Pimenidis, E. (Eds.), Artificial Intelligence Applications and Innovations, Springer International Publishing, Cham. pp. 244–255. doi:10.1007/978-3-030-49186-4_21.

[25] Seboka, N., 2020. Regression model development for showing relation between mechanical yarn stretch (%) in sizing and warp yarn breakage (cmpx) in looms using anova model. Ethiopian journal of textile and apparel 1. URL: https://ssrn.com/abstract=3791327.

[26] Yao, G., Guo, J., Zhou, Y., 2005. Predicting the Warp Breakage Rate in Weaving by Neural Network Techniques. Textile Research Journal 75, 274–278. doi:10.1177/004051750507500314.

[27] Zhao Bo, 2010. The prediction of warp breakage rate of weaving by considering sized yarn quality using artificial neural network theory, in: 2010 International Conference On Computer Design and Applications, IEEE. pp. V2–526–V2–529. doi:10.1109/ICCDA.2010.5541337.

Changes Made after Revision

- **"A few typo and mistakes like in page 4. f_t , N and t are variable that are not linked to any function." –** We altered the IDF function and subsequent text to match the equation variables. Also we corrected the typos in the manuscript.
- **"subsection3.1: are authors referring to columns or rows?" –** We have rewritten the text in order to make it more clear and altered table 1 to include a better context.
- **"The authors rely on AutoML without any explanation. Why using these approches knowing they require parameter optimization?"–** We now detail this question by adding the following text in section 3.3: [The implementation of the AutoML procedure automatically selects the best among several state-of-the-art ML algorithms and allowed us to better focus on feature engineering, which is a non-trivial task in this domain.]
- "**The authors stops optimization after 1 hour, how do you ensure the parameters are optimal."**– We now detail this comment by adding the following text in section 3.3 : [These AutoML tools provide a performance based stopping criterion, which will stop the search process when the performance does not improve by a specified amount. In our case, we assumed the default baseline performance criterion for each AutoML tool.]
- **"I am not sure why the a autokeras has not been used to develop three separate NN for each fault? How can the developed NN predict three different faults with one single model? are faults somehow linked?"–** We have rewritten the text in order to make it clearer. We have added the following text in section 3.3 : [Considering that the same fabric can have all three faults in a single production order, this research compares two main ML approaches: single target and multi-target.]