

KBS-Weld - cloud based platform for design of the welding procedure specification

A. C. Murariu¹, A. G. Cuetos², J. A. Sánchez del Rivero²

¹ National Research & Development Institute for Welding and Material Testing - ISIM Timișoara, Romania

² Izertis S.A., Gijón, Spain

E-mail: amurariu@isim.ro

Keywords

Artificial intelligence, knowledge, data processing, machine learning, cloud, welding, WPS

1. Introduction

In all welding-related industrial fields, welding standards and customer requirements are more and more demanding. Welding procedure and personal qualification required accuracy and could be very time-consuming, and all of these is taking a lot of resources. To support the manufacturers of welded structures a lot of software were developed and is present on the market. Some of them manage welders and procedures data [1, 2] and other help welding engineers to edit the Welding Procedure Specifications (WPS) [3] as they have built-in databases with parent materials and filler materials, they contain forms that support entering required information to create the WPS, and even performs some calculus (e.g. heat input and carbon equivalent). All of these are useful tools for engineers, but none of them can automatically generate a preliminary Welding Procedure Specification (pWPS) without the user intervention. The current paper presents the KBS-Weld cloud-based platform that aims to serve as reference point for professionals of the welding industry where they can share knowledge. Based on such knowledge in form of data related to materials, experiments, test results, welding techniques, parameters etc, the system can exploit the data generated by means of advanced machine learning techniques and can provide optimal combination of materials and welding parameters for welded structures meeting the requirements provided by the end-users.

2. System construction and integration

For building of the computational knowledge-based system it was assumed that it has to determine whether a specific configuration of welding parameters is valid for a given weld. This decision is made based on the estimation of the results of different destructive tests such as tensile tests, bending tests, hardness tests or occurrence of the imperfections that are treated in ISO 5817 and were highlighted using non-destructive tests (NDT). All estimates are data-driven, that means, they are made based on the analysis of historical data. In order to automatize this estimation process, techniques such as the designing of factorial experiments, advanced pattern recognition methodologies, supervised learning and rule-based systems were used.

To construct the KBS-Weld system, specific machine learning algorithms were developed to estimate or predict, based on historical data, types of imperfections that can occur in a weld and its dimensions, based on a set of welding parameters.

Machine learning algorithms learn from observations that occurred in the past (in the case of this project, configuration parameters, sample descriptors, material properties, test results etc.). During this task, the information of past cases has been modelled in the form of a decision vector, that is, each welding test performed in the past has been represented as a set of attribute-value pairs and a target variable to be predicted. These variables can be numerical (e.g. current intensity, current voltage, travel speed), categorical (type of joint), or binary.

Once a starting point dataset has been available, the exploratory data analysis, variable selection and pre-processing phases were performed. In the analysis phase, statistical techniques have been used to identify the subset of the most significant variables that allow describing the models to be learned. Further, data pre-processing phase was performed, being responsible for cleaning and enriching the records. Once the important variables have been identified, in this stage the records with noise that could distort the training phase have been deleted. In turn, information derivation techniques have been used to complete those variables that contained erroneous information (without value or with an invalid value) in some records.

Next, some machine learning algorithms have been tested:

- Linear models [4]: Effective and simple models that generate an output as a function of the inputs. In this case linear regression and logistic regression have been used (fig. 1).

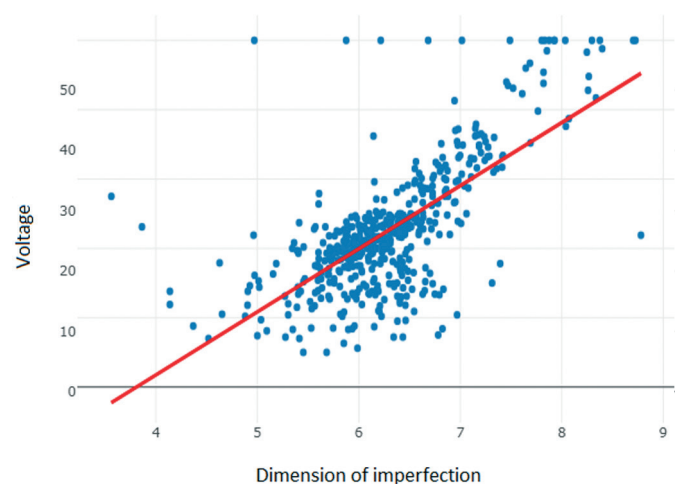


Figure 1. Simple Linear Regression

- Tree-based methods: These are models where the target variable is a continuous value; and observations on an event are mapped to conclusions about the target value of the event. They model nonlinear relationships very well and are lightweight models that offer great performance. In this task, Decision

Trees, Random Forest (fig. 2) and Gradient Boosting [5] based on Decision Trees have been used both in their classification and regression version (fig. 3).

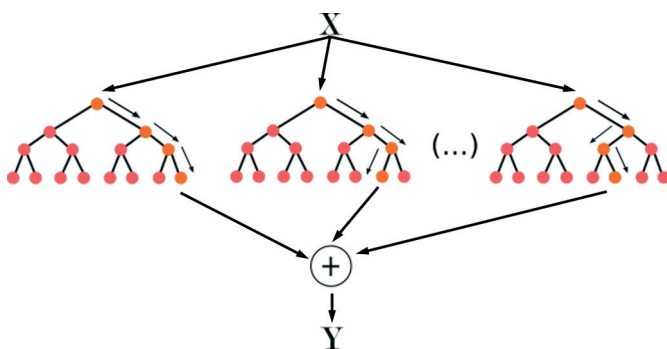


Figure 2. Random Forest

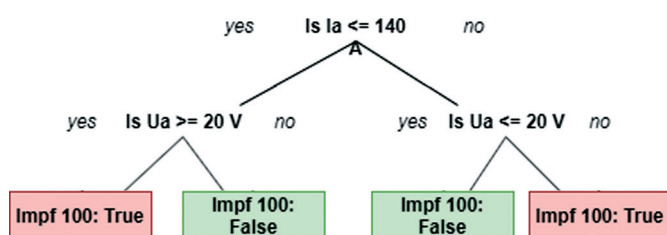


Figure 3. Tree-based model

- Kernel methods [6]: These methods implement functions, called kernels, that transform the data to a much larger dimension to convert a nonlinearly separable problem to a linearly separable one (fig. 4).

In order to validate the previous models, a technique called cross-validation [7] has been used (fig. 5). This technique allows us to measure the performance (in an exhaustive manner) of the algorithms in terms of RMSE (Root Mean Squared Error), MAE (Mean Absolute Error) and MAPE (Mean Percentage Error) for the regression models or F1-score for the classification models.

Once the models that predict the type of imperfection and their dimension from a welding configuration, are generated,

the output of these models is provided as input to the rule-engine system. This system is responsible for interpreting of the ISO 5817 [8], which establishes criteria quality for fusion welding of steels, nickel, titanium and their alloys, and determine if a welding configuration is going to be accepted or not (fig. 6).

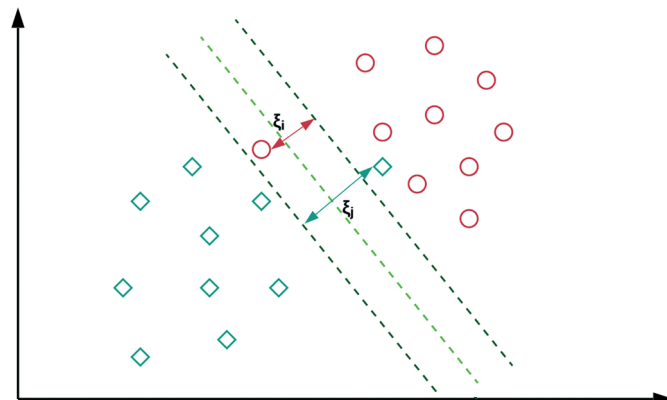


Figure 4. Kernel Method (SVM)

Iteration 1	Test	Train	Train	Train	Train
Iteration 2	Train	Test	Train	Train	Train
Iteration 3	Train	Train	Test	Train	Train
Iteration 4	Train	Train	Train	Test	Train
Iteration 5	Train	Train	Train	Train	Test

Figure 5. Cross-validation graphically explained

The intelligence module has been programmed using the Python programming language and has been deployed as a service, accessible through a REST API (fig. 7).

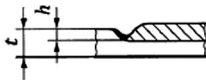

No.	Reference to ISO 6520-1	Imperfection designation	Remarks	t mm	Limits for imperfections for quality levels		
					D	C	B
1 Surface imperfections							
1.1	100	Crack		≥ 0,5	Not permitted	Not permitted	Not permitted
1.2	104	Crater crack		≥ 0,5	Not permitted	Not permitted	Not permitted
1.3	2017	Surface pore	Maximum dimension of a single pore for — butt welds — fillet welds	0,5 to 3	$d \leq 0,3 s$ $d \leq 0,3 a$	Not permitted	Not permitted
			Maximum dimension of a single pore for — butt welds — fillet welds	> 3	$d \leq 0,3 s$, but max. 3 mm $d \leq 0,3 a$, but max. 3 mm	$d \leq 0,2 s$, but max. 2 mm $d \leq 0,2 a$, but max. 2 mm	Not permitted
1.4	2025	End crater pipe		0,5 to 3	$h \leq 0,2 t$	Not permitted	Not permitted
				> 3	$h \leq 0,2 t$, but max. 2 mm	$h \leq 0,1 t$, but max. 1 mm	Not permitted
1.5	401	Lack of fusion (incomplete fusion)	—	≥ 0,5	Not permitted	Not permitted	Not permitted
		Micro lack of fusion	Only detectable by micro examination	≥ 0,5	Permitted	Permitted	Not permitted
1.6	4021	Incomplete root penetration	Only for single side butt welds 	≥ 0,5	Short imperfections: $h \leq 0,2 t$ but max. 2 mm	Not permitted	Not permitted

Figure 6. Limits for imperfections (examples from ISO 5817)

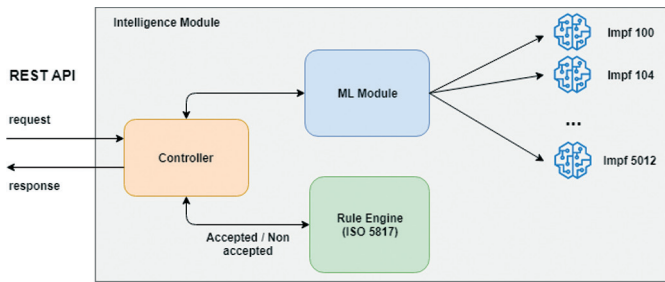


Figure 7. Intelligence module architecture

To optimize the software integration work, DevOps practices [9] were applied. It allows to deliver the software quickly and reliably, as well as allowing scalability and resolution of incidents efficiently. Thus, by automating and monitoring the construction of the software in all its phases, from the integration of code from different developers, to the testing and deploy stages, as well as the management of the infrastructure and tools necessary for its development and production, creation of workflows were enabled, that result in the efficiency and speed.

On the other hand, one of the main points to achieve the objectives set by the DevOps methodology is the coordination of the work carried out by the team of developers. Thus, the concept of continuous integration has been implemented to coordinate and integrate the work of the entire software development team on a main line very frequently. With continuous integration, problems arising from a team of developers working simultaneously on the same project are avoided. To do this was strictly necessary to define the tasks in a concise manner and trying to divide and encapsulate the changes so that they can be integrated regularly to the main line so that the rest of the team can synchronize their work with them.

To implement continuous integration system, various platforms were used. Thus, thanks to the use of GitLab [10] (as a control of source code versions) and Docker, the delivery is immediate once the tests have been executed correctly when doing the automatic deployment on the servers so that a minimum time elapses between a functionality is developed and when it can be used. On the other hand, through Bamboo system [11] we have managed the continuous integration, test automation and deployments. Through Bamboo, the following tasks are performed: definition of compilation plans, artefact generation (Wars, Zip, Docker images, etc.), definition of deployment plans, release management. By Sonar [12] tool revision of the source code was performed which allows analysis of different languages, vulnerability review, coverage analysis and definition of quality gates.

3. Graphical user interface (GUI)

The KBS-Weld platform graphical user interface (GUI) has a proper designed to allow the users to enter data and to interrogate the system database to create proper pWPS (preliminary Welding Procedure Specification). After the login process, users can see the welcome screen as shown in fig. 8. In the left side of the screen is the navigation menu to make it easier to move around the site.

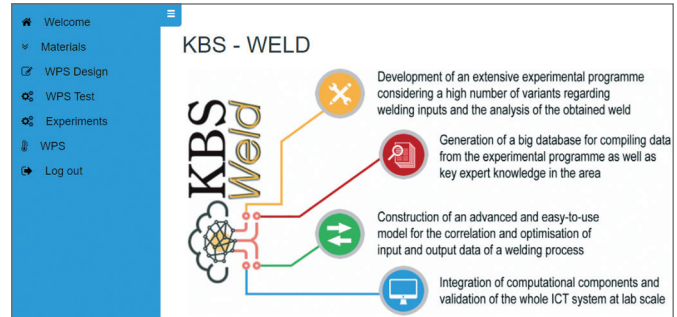


Figure 8. Welcome screen

If the user browses the web application and clicks on “Materials >> Base Materials”, the application displays a list of base materials described by their standard, steel grade, group number and the equivalent carbon number (fig. 9). Users can edit or delete such base materials. On the other hand, an option to search a base material by standard, steel grade or group number is available. There is also an option to add a new base material by clicking the bottom in the upper right side of the screen.

In order to add a new base material to the platform, users have to input information related to the designation, group number etc and also information related to chemical composition and mechanical properties.

The filler materials and gases screens work in a similar way as the base materials screen and they are shown in fig. 10.

When the user browses the “Experiments” tab in the navigation menu, the application displays the list of experiments already uploaded on the platform. Users can search for experiments by features and edit or delete own experiments data. By clicking on the bottom “New Experiment” users can upload new experimental data.

By clicking on “WPS”, in the navigation menu, users can access the list of WPS available in the platform. Following the same schema as the other sections, users can search for WPS by features and edit, delete or add a new WPS.

The aim of WPS Design is that, based on the mechanical and structural characteristics of the materials used during the welding process, as well as other initial conditions (e.g. material thickness,

Standard	Steel Grade	Group Number	CEV2	Actions
1.5415	16Mo3	1.1	0.3293	
1.0345	P235GH	1.1	0.4067	
1.0111	P245NB	1.1	0.21	
1.0425	P265GH	1.1	0.48	
1.0423	P265NB	1.1	0.2567	

Figure 9. Base materials

welding type – selected by the users (fig. 11), the system provides an optimal WPS. In order to do that, first, the users must provide some information related to the weld they want to perform. Then, the system tries to find in the database the WPS that match the information inputted by the user. If there is no WPS in the database that meets the input parameters, the system provides a pWPS based on the existing knowledge in the system. Finally, when a WPS or pWPS is found, the system displays it and allows the user to download the preliminary welding procedure specification.

In the “WPS Test” screen, the system displays a list of tests already performed by the user and it provides the option of creating a new test. In this functionality, based on the material selected, initial configurations and welding parameters inputted by the user, the system will provide an estimation of the mechanical properties of the weld and, also, whether the weld meets the quality requirements established in the ISO 5817.

4. Conclusions

The aim of the KBSWeld platform is to serve as reference point for professionals of the welding industry where they can share knowledge. Knowledge in form of data related to materials, experiments, test results, welding techniques and parameters etc. It has been proved that based on such knowledge, exploiting the data generated by means of advanced machine learning techniques, the system can provide optimal combination of materials and welding parameters for welded structures meeting the requirements provided by end-users. Subsequently, the KBSWeld system helps reducing cost related to previous experimentation and tests performed before producing WPS.

KBS-Weld cloud-based platform meets the requirements of performance, availability, reliability, scalability and security, and it was built based on independent modular components, thus facilitating the integration of more modules ensuring flexibility and adaptability of the solution.

Acknowledgements

This work was carried out as part of KBS-Weld project, supported by the European Union research program COFUND MANUNET III, co-financed by the Romanian Executive Agency for Higher Education, Research, Development and Innovation Funding (UEFISCDI). The authors would like to acknowledge the support provided by the IZERTIS SA, by the National R&D Institute for Welding and Material Testing - ISIM Timisoara and by the SC SAM Robotics SRL, for all the facilities necessary to implement the experimental research.

References

- [1]. Weld assistant: <https://www.weldassistant.com/>;
- [2]. Weldnote: <https://www.youtube.com/watch?v=FWXym98j-6g>;
- [3]. Weld pulse: <https://www.youtube.com/watch?v=B0PR9m1O6d4>;
- [4]. Regression diagnostics: Identifying influential data and sources of collinearity, by D. A. Belsley, K. Kuh and R. E. Welsch;
- [5]. “Lightgbm: A highly efficient gradient boosting decision tree”, G. Ke et al., Proc. Adv. Neural Inf. Process. Syst., pp. 3149-3157, 2017;
- [6]. Cortes, Corinna, and Vladimir Vapnik. “Support-vector networks.” Machine learning 20.3 (1995): 273-297;
- [7]. Stone M. Cross-validatory choice and assessment of statistical predictions. J. Royal Stat. Soc., 36(2), 111–147, 1974;
- [8]. ISO 5817 Welding – Fusion-welded joints in steel, nickel, titanium and their alloys (beam welding excluded) – Quality levels for imperfections;
- [9]. <https://en.wikipedia.org/wiki/DevOps>;
- [10]. <https://gitlab.com>;
- [11]. <https://www.atlassian.com/es/software/bamboo>;
- [12]. <https://www.sonarqube.org/>.

Figure 10. Filler materials

Figure 11. WPS Design Joint Detail