International Journal of Human–Computer Interaction

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/hihc20

The Use of Responsible Artificial Intelligence Techniques in the Context of Loan Approval Processes

Erasmo Purificato, Flavio Lorenzo, Francesca Fallucchi & Ernesto William De Luca

To cite this article: Erasmo Purificato, Flavio Lorenzo, Francesca Fallucchi & Ernesto William De Luca (2022): The Use of Responsible Artificial Intelligence Techniques in the Context of Loan Approval Processes, International Journal of Human–Computer Interaction, DOI: 10.1080/10447318.2022.2081284

To link to this article: https://doi.org/10.1080/10447318.2022.2081284

Published online: 12 Jun 2022.

INTERNATIONAL JOURNAL OF HUMAN–COMPUTER INTERACTION
https://doi.org/10.1080/10447318.2022.2081284

The Use of Responsible Artificial Intelligence Techniques in the Context of Loan Approval Processes

Erasmo Purificato[a,b], Flavio Lorenzo[c], Francesca Fallucchi[b], and Ernesto William De Luca[a,b]

[a]Otto von Guericke University, Magdeburg, Germany; [b]Leibniz Institute for Educational Media, Georg Eckert Institute, Brunswick, Germany; [c]Cafcom S.r.l., Brescia, Italy

ABSTRACT

Despite the existing skepticism about the use of automatic systems in contexts where human knowledge and experience are considered indispensable (e.g., the granting of a mortgage, the prediction of stock prices, or the detection of cancers), our work aims to show how the use of *explainability* and *fairness* techniques can lead to the growth of a domain expert's trust and reliance on an artificial intelligence (AI) system. This article presents a system, applied to the context of loan approval processes, focusing on the two aforementioned ethical principles out of the four defined by the *High-Level Expert Group on AI* in the document "Ethics Guidelines for Trustworthy AI," published in April 2019, in which the key requirements that AI systems should meet to be considered *trustworthy* are identified. The presented case study is realized within a proprietary framework composed of several components for supporting the user throughout the management of the whole life cycle of a machine learning model. The main approaches, consisting of providing an interpretation of the model's outputs and monitoring the model's decisions to detect and react to unfair behaviors, are described in more detail to compare our system within state-of-the-art related frameworks. Finally, a novel *Trust & Reliance Scale* is proposed for evaluating the system, and a usability test is performed to measure the user satisfaction with the effectiveness of the developed user interface; results are obtained, respectively, by the submission of the mentioned novel scale to bank domain experts and the usability questionnaire to a heterogeneous group composed of loan officers, data scientists, and researchers.

1. Introduction

The ability to predict the occurrence of certain events in advance has always been a critical factor in the financial and banking fields (Board, 2017; Heaton et al., 2016). An estimation of the risk associated with the granting of a loan by a banking institution requires deep expertise and long experience on the part of loan and credit officers exploiting information related to the customers, concerning their personal data, financial situation and credit history, as well as the entity of the specific request made. Quoting the study of DataRobot (2019), "*Today, you would be hard-pressed to identify a line of business or function in a bank that does not have multiple needs for predictive analytics*" and the amount of data required for the predictive analysis in money lending, to which past information about granted loans must be added, make this one of the most interesting application fields for artificial intelligence (AI) techniques of the whole banking sector.

In recent years, machine learning (ML) models have been used to predict stock prices (Hagenau et al., 2013; Schumaker & Chen, 2010), to identify the presence of cancers (Hirasawa et al., 2018), or as in the case study presented in this article, to decide whether to grant a loan to bank customers (Arun et al., 2016). Given the specific field of application, the risk associated with the prediction being computed may vary significantly. As reported in several studies (Goebel et al., 2018; Holzinger et al., 2017), although AI systems are equaling, or even exceeding human performance, in many fields, their use is still viewed suspiciously and the human experience is considered irreplaceable (Jarrahi, 2018). There are contexts where understanding motivations leading to a specific result is more important than the result itself, and it is crucial to be able to understand the reasons why a prediction was made to build *trust* in the decisions taken by a model. Trust is one of the key factors that influence the adoption of ML techniques inside high-risk applications and brought about the rise of the fields of study called *explainable artificial intelligence* (XAI) and *responsible artificial intelligence* (RAI).

Due to this ubiquity, concerns are starting to arise about whether the development of AI systems, and the decisions made by them, should be based on a set of *ethical principles* to promote transparency, social equity, sustainability, and avoid social injustices. In particular, considering our case study on automatic predictions in loan approval processes, several critical elements in the European law have to be taken into account when individuals are assessed by such an

CONTACT Erasmo Purificato erasmo.purificato@ovgu.de Otto von Guericke University, Magdeburg, Germany

algorithm (Commission, n.d.; Goodman & Flaxman, 2017): their rights to not be subject to an automated decision in the first place, their right to get an explanation of the decision and their right to non-discrimination. As well-reported in the article written by Angel Perez for *2021.AI*, "*Fairness in Machine Learning*,"[1] ML practitioners should develop models that, *by design*, take care of possible discriminations and that are explainable to users, requiring high transparency and reproducibility throughout the whole ML workflow.

A significant contribution in this direction has been provided by the *High-Level Expert Group on AI* (AI-HLEG, 2019), appointed by the European Commission, that presented the document "Ethics Guidelines for Trustworthy Artificial Intelligence." As the guidelines' authors note, the concept of *trustworthy AI* is made of three main components: compliance with existing laws and regulations (*lawful AI*); alignment with society's ethical principles, even in those situations in which no regulation has been developed yet (*ethical AI*); robustness both from a technical and social perspective to avoid incorrect behaviors that may cause unintentional harm (*robust AI*). The AI HLEG group identifies four ethical principles that must be satisfied for an AI system to be considered trustworthy: *respect for human autonomy*, *prevention of harm* to other human beings, *fairness* of the AI system's decisions, and *explicability* of the outcome of an AI system.

According to the guidelines, AI systems should provide clear explanations for their outputs, and the way a system interacts with a user should never be interpreted as though the decision were made by a human rather than a machine. Explainable AI has been defined by Gunning (2017): "*XAI will create a suite of machine learning techniques that enables human users to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners*." Extending this concept to include principles like fairness, we can refer to *responsible AI*, the definition of which has been provided by Dignum (2018, 2019): "*Responsible AI is about human responsibility for the development of intelligent systems along with fundamental human principles and values, to ensure human flourishing and well-being in a sustainable world*."

Within the AI community, the debate about the need for explainability is very heated. For instance, G. Hinton considers the constant search to explain how an AI system works a "*complete disaster*" (Simonite, 2018). Our point of view is opposite to that, and according to Miller (2019b), we consider explainability significant for two main reasons: *trust*, because people cannot just read some statistics about the model performance and believe a decision is correct, and *ethics* because we have to prove that a developed system is not producing discrimination of any kind. Thus, a successful XAI, or more specifically, RAI system, must relate to social sciences (Miller, 2019c). The evaluation of *explainable user interfaces* (UIs) is another crucial topic. Even though the satisfaction of users interacting with systems providing explanations (in contrast with systems without them) can be seen as an easy and predictable outcome, we believe a concrete evaluation is always needed, especially in the domain where dealing with explainable UIs is not a daily routine. Our claim is also supported by some evidence in literature: Millecamp et al. (2019) showed that in certain contexts and for specific users, explanations could even create a lack of confidence in the system; Wang et al. (2022) showed that users could prefer a biased model instead of an unbiased one, in case of the lack of proper result explanations.

Linked to explainability and ethics is the concept of *fairness*. The capability to understand how a prediction was made can help expose the ML model's discriminatory behavior, thus allowing detecting and mitigating biases deriving from the examples provided by humans as the foundation on which these models are built. As a result, predictions made by these systems may favor a majority group over some minorities.

In this regard, a canonical example comes from the *COMPAS*[2] algorithm, which several courts use in the U.S. as a risk assessment tool to estimate the probability of a person committing another crime. Based on the algorithm prediction, judges use COMPAS to decide whether to release an offender. An analysis published by ProPublica (Angwin et al., 2016) has highlighted even more the problem within the scientific community (Washington, 2019), demonstrating how the algorithm was unfairly judging black offenders, which were wrongly labeled as high-risk individuals at almost twice the rate as white defendants.

Strictly related to the desire of building a valuable, trustworthy AI system, there is the need to develop an understandable and easy-to-use *user interface* (UI), which is currently one of the weak points of research in the XAI field (Abdul et al., 2018).

The scope of our work is to show how the use of explainability and fairness techniques can lead to the growth of a domain expert's trust and reliance on AI systems. With this aim, four functionalities are proposed: a *dataset & ML model handler*, a *standardized explainability tool*, a *fairness tool*, and a *feedback loop*. The first of these components allows users to load a dataset and preprocess it, training several ML models to find the most performing one on the provided dataset and monitor its performance. The *standardized explainability tool* provides methods to get explanations for each prediction; since our goal is to build a tool to support the creation of a responsible AI system regardless of the specific ML model it is based on, only *model-agnostic* solutions are employed. The *fairness tool* grants users the ability to detect biases within the model's behavior through a proposed algorithm based on *disparate impact* metrics and mitigate them using the *reweighing* algorithm that follows the *independence* criterion, one of the criteria that has legal support. An *unbiased* version of the original model can be trained at the end of the described procedure. The *feedback loop* allows a domain expert to judge the model's results with the related explanations to create a new ground truth to train a new, more performing ML model.

We applied the presented system to the context of the *loan approval process* developing a proprietary framework with an intuitive UI and demonstrating its effectiveness through experimental results from field tests and subsequent

user studies. An experimental session for choosing the best explainability algorithm to use in the developed framework is performed following the state-of-the-art *Explanation Goodness Scale*; a novel *Trust & Reliance Scale* is proposed to evaluate the system explainability, while an *A/B test* is carried out for assessing the fairness feature; finally, we performed a *usability test* to evaluate the UI. Results are obtained by submitting the mentioned scales, tests, and the usability questionnaire, respectively, to data scientists and researchers for the explainability algorithms and bank domain experts and loan officers to evaluate the other functionalities.

The article is structured as follows. In Section 2 some of the recent and most relevant research works and platforms published or released in the last years about explainability and fairness are presented. The system approach and the proposed architecture for supporting the whole life cycle of an ML model are discussed, respectively, in Sections 3 and 4. In Section 5 the case study is illustrated. Experimental evaluations are discussed in Section 6. Evaluation scales and questionnaires are reported in Appendix. Conclusions and future work are discussed in Section 7.

2. Related work

This section presents some relevant related research works and commercial platforms regarding our context of interest. Because of our scenario's complexity and heterogeneity, we separately discuss the literature on explainability and fairness. From the analysis of the following state of the art, we have derived the foundations of our system's approach, which is based on the limitations of current XAI systems.

2.1. Explainability

As the amount of data being collected grew and ML models became faster and more accurate, their applications have extended to several different fields and stakeholders (Preece et al., 2018). This is why it is becoming crucial to be able to explain the reasons behind any predictions.

Comprehensive surveys about concepts, taxonomies, issues, methods, and challenges in the fields of explainability and XAI have been provided in the last three-year period (Adadi & Berrada, 2018; Barredo Arrieta et al., 2020; Biran & Cotton, 2017; Gilpin et al., 2018; Miller, 2019a; Mueller et al., 2019).

In recent years many algorithms have been designed to provide insights into which features are most likely to influence the model predictions. They can be divided into two categories: *model-specific solutions* and *model-agnostic solutions*. The first one is tailored to specific model classes. They can provide further insights into a model prediction by exploiting the specificities of the model class of interest, both for *shallow* ML models, such as ensemble classifiers (Palczewska et al., 2014; Rajani & Mooney, 2018) and SVMs (Landecker et al., 2013), and *deep* ML models, such as multi-layer neural networks (Shrikumar et al., 2016) and convolutional neural networks (Selvaraju et al., 2017). Algorithms belonging to the second category aim to be applied to any ML model. The strategy beyond these techniques consists in considering the model as a black box, and they work by analyzing only the input features and the model output. Relevant contributions to this category are *LIME* (Ribeiro et al., 2016b) and its variants (Ribeiro et al., 2016a), *SHAP* (Lundberg & Lee, 2017), based on the concept of the Shapley values derived from games theory (Shapley, 1953), and *Anchors* (Ribeiro et al., 2018).

Most of the commercial platforms developed in the XAI field exploit at least one of the aforementioned techniques, sometimes improving them to achieve better results. Usually, they can be helpful to data scientists and researchers, but less for non-expert users. Some popular platforms and frameworks are briefly presented in the following.

IBM Watson OpenScale[3] is a commercial solution belonging to the IBM Cloud suite, introduced to provide a platform that could be used by businesses to operationalize their AI systems and to extend their deployments to the whole enterprise. It offers several tools that help data scientists and managers monitor and understand their model's outcomes. OpenScale not only provides an online application to navigate through the results employing a graphical user interface, but it also offers an API[4] that allows accessing the platform's services programmatically. It currently provides two different explainability techniques: *LIME* (Ribeiro et al., 2016b) and a variant of *MACEM* (Dhurandhar et al., 2019). While OpenScale does not directly provide the capability to manage the user's datasets and train custom models, these tasks are achieved by integrating OpenScale with the rest of the IBM cloud services. Indeed, the suite to which OpenScale belongs includes other tools that assist the user during all the steps related to developing a custom ML model. Several factors might limit the application of a commercial solution, such as OpenScale. The first one, and in many cases the most crucial argument against OpenScale, is its expensiveness (although the cost comes with product support, documentation, and bug fixing, as well as brand reliability, it is not adequate for a bank institution that has not its core business in AI). Other issues are related to the lack of control over what happens inside OpenScale, and the requirement to have the training data stored on the IBM Cloud platform if a user does not want to choose their hybrid cloud solution, named *IBM Cloud Pak for Data*, that provides OpenScale on customer's machines.

Google What-If Tool[5] (WIT) is an interactive tool that allows users to investigate the model's behavior and performance through a visual interface. This tool is an initiative of Google's *People + AI Research* (PAIR) team. It has been proposed to enable people to evaluate machine learning models without the need to write complex code. Through WIT, a user can investigate the behavior of multiple models, compare them, and extract insights from them. The visual approach followed by WIT leverages the predictions obtained from a test set to provide customizable graphs which offer a better understanding of the relationship between different attributes and the predicted label. The tool is also helpful for comparing different instances and observing how each

prediction changes as the values of its attributes are modified. While WIT is excellent for a data scientist trying to understand his model better, domain expert users wish for a straightforward explanation about why a specific result has been obtained.

Google Cloud Explainable AI[6] is a set of tools and frameworks developed to aid data scientists to build interpretable ML models. With it, developers can understand feature attributions in *AutoML Tables* and *AI Platform* (the Google Cloud Platform catalog provides both tools) and visually investigate model behavior using the *What-If Tool*. It also simplifies model management using the AI Platform. Google Explainable AI leverages the integration of Google Cloud's AI Explanations service into AI Platform Prediction to provide *feature attribution*. As for IBM Watson OpenScale, it needs other services from Google Cloud Platform (GCP) to make users able to manage the whole life cycle of an ML model. GCP's AI Explanations offer three methods to use for feature attributions: *sampled Shapley* (Maleki et al., 2013), *integrated gradients* (Sundararajan et al., 2017), and *XRAI* (Kapishnikov et al., 2019). Each of the mentioned methods is based on Shapley values, and users can select what they prefer for their explanations requests.

AI Explainability 360[7] is a good open-source software toolkit addressed to different stakeholders, from domain experts to system developers. It allows exploring eight state-of-the-art explainability methods and two evaluation metrics. Noteworthy is the provided effective taxonomy to help to navigate the space of explanation methods, not restricted to those in the toolkit, but also in the literature (Arya et al., 2019).

Several XAI research works presented in different fields, such as HCI (Abdul et al., 2018; Zhu et al., 2018), visual analytics (Tamagnini et al., 2017) and medicine (Holzinger et al., 2017; Lamy et al., 2019) denote that most researchers are focusing on new algorithms, and not on usability or efficacy effective UIs understandable by non-expert users.

2.2. Fairness

Defining a *fairness criterion*, to evaluate a system to consider it as fair, is a complex task due to the different ways biases can arise. What can be considered fair in a specific context might be unfair in another one. Furthermore, different people have different sensibilities about what is fair and what is not, and what is fair considering individuals or populations as a group (Binns, 2020). Three criteria are commonly contemplated: *independence*, *separation*, and *sufficiency*. *Independence* criterion requires the sensitive characteristic to be statistically independent of the score. Variants of this criterion include *statistical parity* (Dwork et al., 2012), *group fairness* (Friedler et al., 2016; Yeom & Tschantz, 2018), and *disparate impact* (Feldman et al., 2015). *Separation* criterion seeks to acknowledge the existence and rightfulness of the correlation between the sensitive feature and the target variable to the extent that the target variable justifies it. It appears under different names, such as *equivalent odds* (Hardt et al., 2016) and *disparate mistreatment* (Zafar et al., 2017). *Sufficiency* criterion is based on the idea that the sensitive feature is already subsumed in the score used for predicting the target label (Chouldechova, 2017; Kleinberg et al., 2016).

Bias mitigation algorithms are mainly based on two factors: the criterion selected for measuring fairness and the step of the ML process in which it is applied. The steps involved are three: *pre-processing*, *in-processing*, and *post-processing*. In *pre-processing*, the goal is to produce a new representation of the training set in which the information correlated to the set of non-sensitive features is preserved, ignoring the information of the sensitive feature [e.g., *reweighing* (Kamiran & Calders, 2012), *disparate impact remover* (Feldman et al., 2015), *optimized pre-processing* (Calmon et al., 2017)]. The aim of the *in-processing* step is to enforce fairness by changing the learning strategy of the model at training time (e.g., *adversarial debiasing* (Zhang et al., 2018), *disparate impact remover*). The algorithms of the *post-processing* step try to satisfy fairness constraints by slightly modifying the output of a model without the need to change the training data or retrain the model. These algorithms are usually used when the two previous approaches are not viable because the training dataset or the ML algorithm is not accessible [e.g., *reject option classification* (Kamiran et al., 2012), *equalized odds post-processing* (Hardt et al., 2016)].

The application of fairness criteria to ML models is a topic that has received much attention lately, thanks to greater awareness about the risks that unfair AI systems might pose to specific social groups. Relevant research works about the nascent field of Fair ML have been published in the last years (Chouldechova, 2017; Corbett-Davies & Goel, 2018; Holstein et al., 2019).

Not only has this popularity led to a growth in the number of scientific publications related to fairness, but it also encouraged the introduction of several tools whose goal is to monitor the behavior of a model to alert the user in case of unfair treatment. An example is *IBM AI Fairness 360* (AIF360), which is perhaps the most considerable open-source toolkit available for ML fairness. Its goal is to "*examine, report, and mitigate discrimination and bias in ML models throughout the AI application lifecycle*." It is an extensible framework capable of unifying most of the metrics and algorithms presented in this chapter. It also includes a bias explanation feature that gives further insights into the computed metrics (Bellamy et al., 2018).

In addition to the capabilities already presented on explainability, even *IBM Watson OpenScale* gives the possibility to set up a monitor to track the fairness of the model at hand. The presence of biases in the model is estimated based on the *disparate impact* metric. One of the main OpenScale's limitations is that the privileged and unprivileged groups must be selected in advance when setting up the fairness monitor. This operation may become cumbersome as the cardinality of the sensitivity attribute grows. Moreover, the user may not be aware of which value belongs to which group.

3. System approach

Although loan approval processes might benefit from the introduction of automated systems (i.e., ML models) to

support the decision task, several factors have limited their application in this field so far: loan approval processes are high-risk activities that require officers to understand the motivation behind every ML model prediction. It is not enough to demonstrate that a model performs well if considered only a black box. With skeptical users the ability to explain *how* it works, *which* data is important, and *when* is crucial; model decisions have a considerable impact on the future of the customers who applied for the loan, and they must be provided with explanations about *why* their application has been rejected; such a decision must be devoid of biases to ensure that individuals with different origins, cultures, and backgrounds are treated *fairly*.

The main aim of the system presented in this article is to overcome the above challenges by proposing a single solution to create a comprehensive *trustworthy intelligent system* exploiting the principles of *explainability* and *fairness*. In the following of this section, an in-depth presentation of the two concepts is provided, focusing on their relevance in this article.

3.1. Explainability

What do we mean by "*explainability*"? Explainability is considered the core of each AI system that aspires to be considered *trustworthy*, and although it is formally different from *interpretability*, the two terms are considered closely related in literature (Biran & Cotton, 2017). Indeed, interpretability can be considered a static characteristic of a model, referring to the capability to explain its meaning in a human-understandable way. At the same time, explainability is a dynamic characteristic of a model representing actions and procedures beneficial to provide explicit knowledge about why a specific prediction was made using easy-to-understand terms. This means that explainability depends on the people who need to understand the model, so the most appropriate definition could be: "*Given a certain audience, explainability refers to the details and reasons a model gives to make its functioning clear or easy to understand*" (Barredo Arrieta et al., 2020). This property is crucial for building trust in the decisions taken by a model, and it is one of the critical factors that influence the adoption of ML techniques inside high-risk applications. There are situations in which having the capabilities to interpret a result and understand the factors that contributed to it is far more important than having a high-performance model. This is one of the reasons why simpler algorithms, such as *decision trees* and *K-nearest neighbors* (KNN), are widely used despite being less accurate than other options like *neural networks* and *support vector machines* (SVMs).

As our purpose is to develop a Trustworthy AI system for loan approval processes regardless of the specific ML model it is based on, only *model-agnostic* solutions (see Section 2.1) are used in the implementation.

3.2. Fairness

To satisfy the principle of *fairness*, users must be aware of existing biases (i.e., prejudices) that may lead AI systems to discriminate against certain groups of people or individuals. Furthermore, every AI system should be accessible to people of any age, gender, and ability (AI-HLEG, 2019).

So far, no standard definitions of fairness have been drawn up. In the context of decision-making, fairness can be formalized as "*the absence of any prejudice or favoritism toward an individual or a group based on their inherent or acquired characteristics*" (Mehrabi et al., 2019). A recent study about how people perceive fairness in the context of loan allocations (Saxena et al., 2020) shows a preference for a specific definition, named *calibrated fairness* (Liu et al., 2017), that aims to select individuals in proportion to their merit. In particular, that study demonstrates that officers choosing between two loan applicants tend to prefer to split the money in a proportion of their loan repayment rates, instead of an "equal" (50/50) splitting, or giving all the money to the candidate with the higher payback rate. The "ratio" decision is allowed under the calibrated fairness definition.

A machine learning model can be biased because these systems are trained by examples: when using historical data for modeling human behaviors, the provided sample reflects the prejudices of the people who made these decisions in the first place.

The choice of the right bias mitigation algorithm, among those described in Section 2.2, is constrained by several reasons: the definition of fairness may vary from case to case; different criteria cannot be pursued simultaneously and each algorithm mainly focuses on a single definition (the *reweighing* algorithm is based on the *independence* criterion, while the *disparate mistreatment remover* technique uses the *separation* definition); the step of the ML model pipeline in which the user is allowed to intervene: as a rule of thumb, the earlier the algorithms are applied, the most flexible and effective the intervention will be; the requirements of the algorithm itself: for instance, the *equalized odds post-processing* technique, despite being a post-processing strategy, requires access to the sensitive feature to compute the right label. Other algorithms have some limitations in terms of the types of classifiers they can be applied to. Some algorithms, such as *reject option classification*, are deterministic, while others have a randomized component (e.g., *disparate mistreatment remover*).

As detailed in Section 4.3, in the presented system we follow the *independence* criterion and choose the *reweighing* algorithm for bias mitigation. Independence is frequently used in literature because it is one of the few criteria having legal support. The so-called *80%-rule*, or *four-fifth rule*, specified in the *U.S. Equal Employment Opportunity Commission* guidelines, prescribes that a selection rate for any group (classified by race, orientation, or ethnicity) that is less than four-fifths of that for the group with the highest rate constitutes evidence of *disparate impact*, that is, discriminatory effects on a protected group (Biddle, 2006).

In the next section, a top-down description of the system components is provided, from a logical overview up to the illustration of the UI of the developed framework, going

through an in-depth presentation of the explainability and fairness tools.

4. System implementation

This section first provides an overview of the proprietary framework developed for the implementation of the case study. It then illustrates the core of the presented system, that is, the application of *explainability* and *fairness* principles to the context of a loan approval process.

The developed framework provides all the following functionalities to ensure complete management of the ML model life cycle, grouped by their high-level purpose, as shown in Figure 1.

The **dataset & ML model handler** allows users to: load a dataset and store it through a procedure involving a fixed preprocessing step, a custom setup, and a fairness check for preliminary bias detection; find the best ML model to use by training at the same time several models with different algorithms and evaluating them through standard metrics; monitor models performance using various metrics (the same with which a model is evaluated after training).

The **standardized explainability tool** provides users with the ability to get explanations for each prediction to make them (i.e., both *loan officers* and *loan applicants*) able to understand clearly the features (or attributes) that most influence the results, both positively and negatively.

The **fairness tool** provides a fairness check and a bias mitigation process. It can identify the presence of biases within the trained model using one of the state-of-the-art fairness criteria (chosen for the reasons explained in Section 4.3) to make users aware that they are supported or judged by a fair system, with the possibility, otherwise, to retrain a new unbiased version of the same model.

The **feedback loop** allows a domain expert (e.g., the loan officer) to give feedback about a specific prediction to create a new *ground truth* and build a new ML model, hopefully with better performance.

From a conceptual and schematic point of view, the proposed framework is described in the class diagram in Figure 2. A brief description of each class is provided below.

Dataset describes the dataset used to train the model. It is characterized by the number of rows of the dataset and a name used as an identifier. The *id* attribute is also used to retrieve the content of the dataset from the local storage.

Label represents the values that can be assigned to the labels of a dataset. Each possible couple (dataset_id, label_value) is described by an instance of this class. The information about its value and its number of occurrences in the associated dataset is added to distinguish each label.

Model denotes the model trained using a specific dataset. Each model is described by an identifier, a descriptive name, and the date it was added to the system. The Boolean *unbiased* attribute is used to specify if the model was obtained after the application of a bias mitigation algorithm to another model.

MLAlgorithm is used to describe the algorithm with which the model is trained. It is also employed to provide some additional information to the user during the bias mitigation process.

PredictionData is used to represent the prediction computed by a model for a given input instance provided by the user. The outcome of the prediction is stored as the probability returned by the model. The entity also includes, for each attribute of the instance being predicted, its feature value and the related weight (or score) obtained using a *model-agnostic interpretability algorithm* (e.g., LIME or SHAP).

FeedbackData describes feedback provided for a given prediction. It is represented as a Boolean attribute, where *true* means that the prediction aligns with the expectation of the user or with the actual outcome.

4.1. Application workflow

A concise representation of the functionalities provided by the developed framework is shown in Figure 3. Each application flow and its main components are described below. For a better understanding of the diagram, two premises are to be made: components having equal shape, size and name are intended to be the same; they are duplicated only for better visualization; black dashed lines in the diagram represent the connection between the data and the specific processes used.

Loan Approval System User Interface allows users to select the functionality to run through the *Tab menu*. It is implemented in an internet application with an intuitive

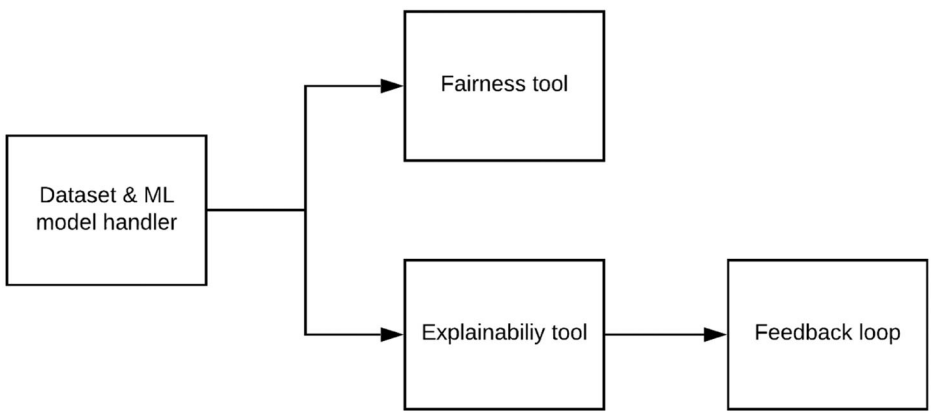

Figure 1. High-level components of the presented system.

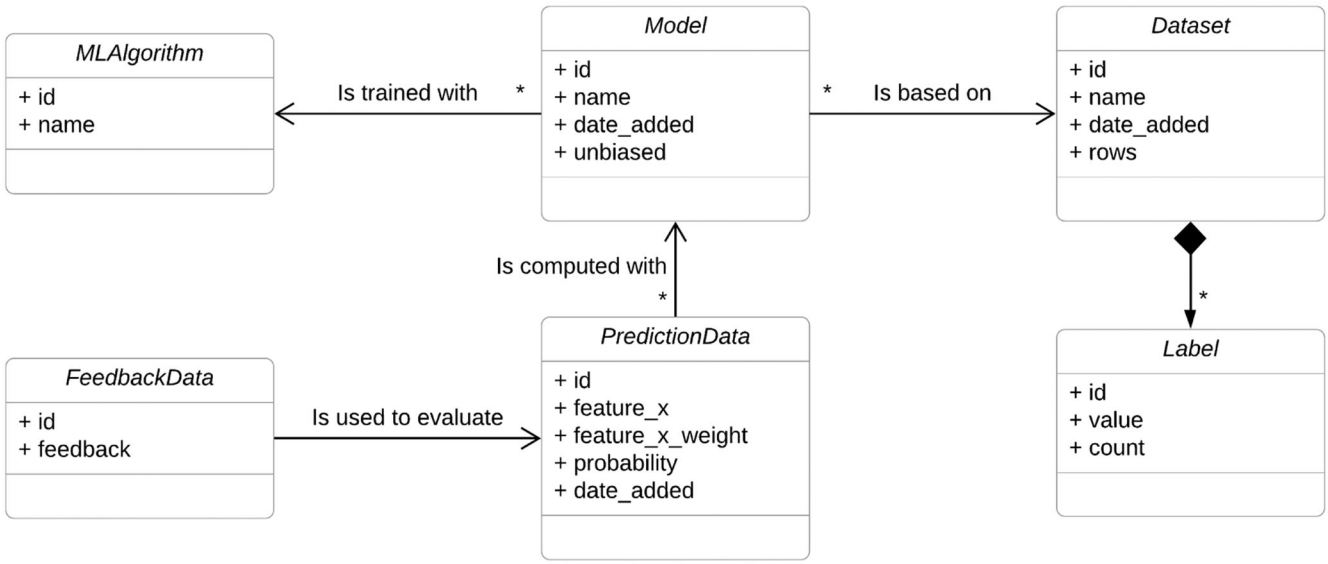

Figure 2. Class diagram of the proposed framework.

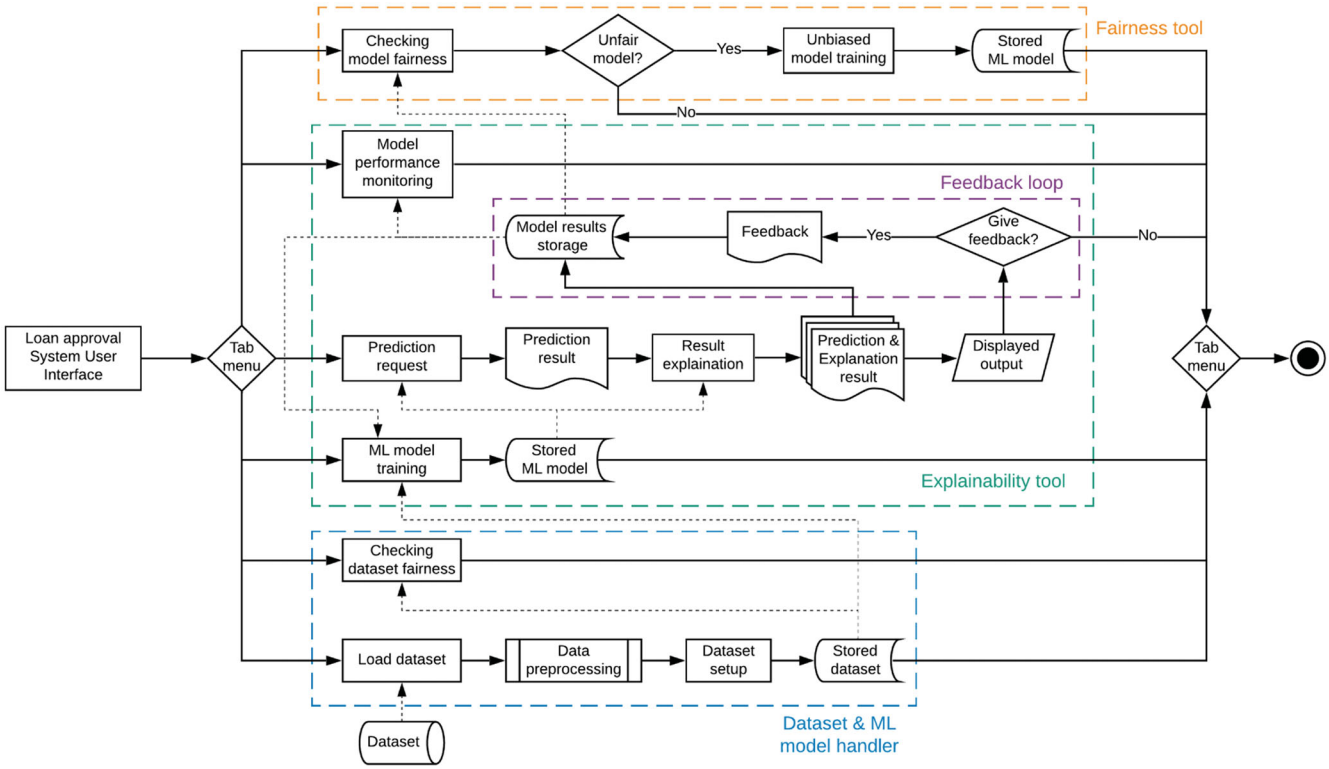

Figure 3. Application workflow.

layout. The application is shortly illustrated below in this section to show how the UI looks to its users.

Load dataset is the functionality to load a dataset and store it in the system. Intuitively, the first time the system starts, this is the only available functionality. Before effectively storing the loaded dataset, a preliminary predefined *Data Preprocessing* is required to prepare the data for the next steps. The *Dataset setup* component allows users to check and modify dataset parameters, including the name to save it under, columns names, and data types. Different datasets can be loaded and stored into the system simultaneously to be selected by users when needed.

Through the **ML model training** functionality, the training phase can be started once a dataset is stored and available for selection. Several models are built at the same time using different ML algorithms. Trained models are presented to users along with metrics (i.e., *Accuracy*, *Precision*, *Recall*, *F1-score*) to evaluate the performance of each of them and compare each other to select the best one to store in the system.

To **request a prediction**, users can select one of the stored ML models and query it to obtain the prediction result and its *explanation*. In our case study, predictions are the probabilities that customers can repay a loan given their

credit histories and some personal data as the input. Once the output is computed, users (domain experts at this point) can *give feedback* about the specific prediction to enable the possibility to monitor the results of the model in use. Predictions, explanations, and feedback are saved in an internal *Model results storage*.

Model performance monitoring functionality exploits the feedback collected from users to compute statistics about the performance of the managed models (same metrics as in the *ML model training*).

Checking dataset/model fairness and bias mitigation process works as follows. Both the original stored dataset and the model in use can be inspected to check the presence of any biases. While checking the dataset is meant for diagnostic purposes to trace back the unfair attribute to the starting labels distribution, the information derived from the model predictions is used to monitor how a sensitive attribute influences the model behavior. If some decisions are considered to be unfair, then an *unbiased model* can be trained and stored.

4.2. The standardized explainability tool

By design, the developed framework provides several ways to obtain the interpretation of a given prediction. Components of this generalized and standardized explainability tool are described below and represented by the diagram depicted in Figure 4.

Configuration class performs the preprocessing steps required to use the explainability algorithms and to train explainer models. Starting from the dataset, categorical and numerical features are extracted to be evaluated, and the *one-hot encoding* procedure is applied to categorical values. The *configuration* class is also used to prepare the instance to be explained for the application of the interpretability algorithms by exploiting the data extracted during the mentioned initialization phase.

Explainer interface is the common interface that standardizes the access to the different interpretability algorithms and allows to switch from one explainer to another smoothly or to use different explainers at the same time. Each *explainer* is initialized using the *configuration* class previously described and their explanations are generated through the compute_explanation method. Depending on the specific implementation, the resulting output can be returned in different formats (e.g., lists or dictionaries).

Three different implementations of the interface illustrated above are possible in the presented framework. Each of these classes, listed in the following, leverages some other services to work correctly; initialization and formatting of the requests to these external packages are handled by the *explainer* and *configuration* components.

LimeExplainer produces a score for each feature based on the *LIME* algorithm (Ribeiro et al., 2016b), but normalizes it so that, by summing up all the scores, the total value amounts to 100. This operation is made to avoid a misunderstanding of LIME results.

ShapExplainer is a wrapper around the *SHAP* algorithm implementation (Lundberg & Lee, 2017). Its output is handled to produce a result similar to the one provided by the *LimeExplainer*.

AnchorsExplainer is based on the *Anchors* algorithm (Ribeiro et al., 2018) and, as in the previous ones, leverages the implementation proposed by the article authors and provides a standardized output score.

The standard process of the explainability tool is shown in Figure 5. First, a *configuration* object is created by the service layer using the original training dataset. Second, the generated instance is used to initialize an *explainer* object. In the third step, the service layer requires the new *explainer* object to explain a given instance. Since the actual interpretability algorithm requires the instance to be provided in a specific format, the *explainer* exploits its internal *configuration* to prepare the given instance accordingly. Once the

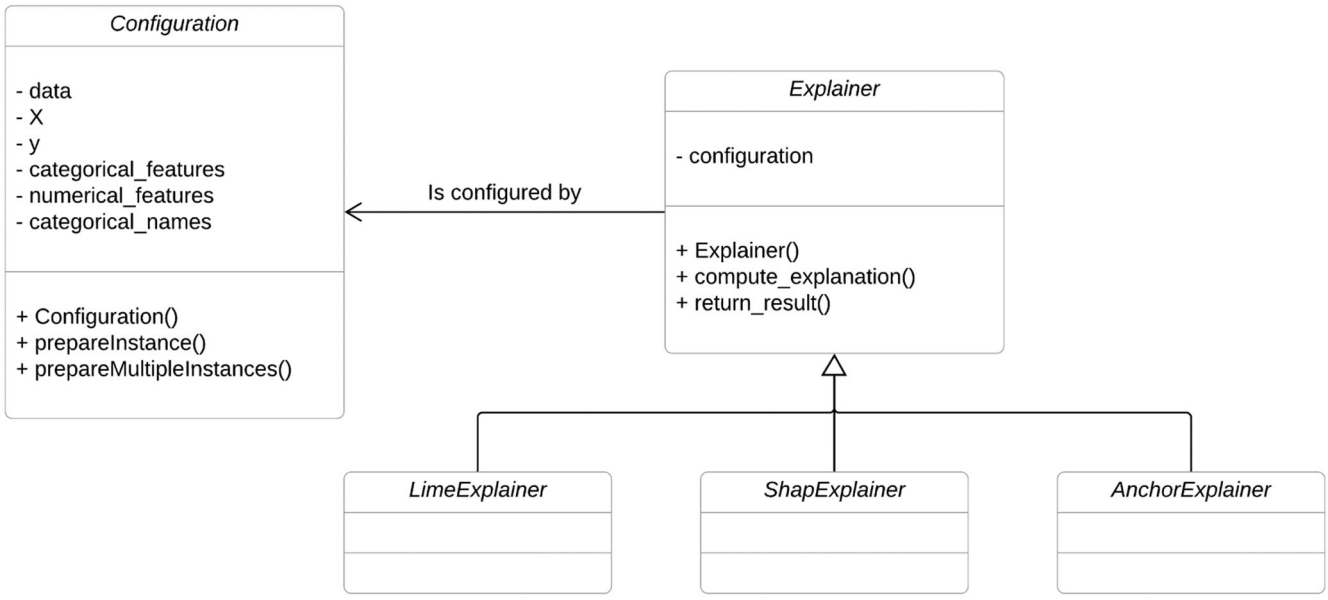

Figure 4. Class diagram of the explainability tool.

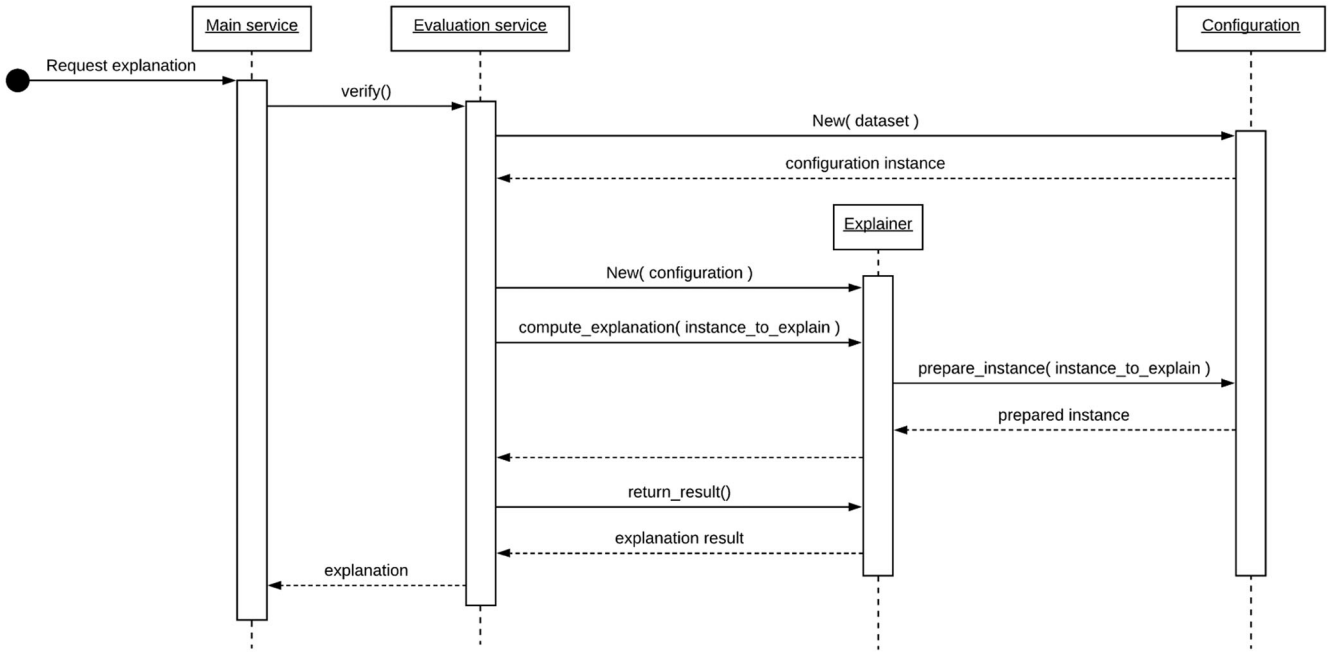

Figure 5. Sequence diagram of the explainability tool.

explainer method has finished its computation, explanations can be retrieved by the service layer.

4.3. The fairness check and bias mitigation processes

The second most important feature the presented system provides is the capability to inspect the original dataset label distributions and the trained models' behavior for bias detection and potentially training an unbiased model. A discussion about these processes follows.

Our system can inspect both the original dataset and models' behavior to determine the lack of fairness. As for the dataset, its rows are used to identify if a bias can be traced back to the original data (this feature is meant for diagnostic purposes); instead, the system uses the predictions made by a model to analyze its behavior. If a bias is detected from the model predictions, then an *unbiased* version of the model can be trained and stored for future predictions. Apart from these differences, the proposed Algorithm 1 describes the *bias detection* procedure for both the original dataset and trained models.

The labeled dataset in input D (which also refers to the data structure in which the results of the model predictions in use are stored) is inspected for biases using the *disparate impact* metric (Feldman et al., 2015). As part of the process, dataset rows $d \in D$ are grouped based on the sensitive attribute value s. These groups are later referred to as *sensitive groups* $g \in G$. For each sensitive group g, positive outcomes ratio is computed, and G is split into one or more *privilege classes* $C \in \mathfrak{C}$, where $C = \{g_i \mid g_i \in G, 0 < i \leq |G|\}$, based on two factors: each privilege class C must represent at least the 5% of the entire population of D, and the disparate impact between C and any other sensitive group g (or vice versa) must be lower than 0.8. The first constraint is applied to guarantee that each privilege class contains a statistically relevant number of instances, while the 0.8 threshold has been selected to comply with the *80%-rule* (Biddle, 2006).

The output of the algorithm is a set of privilege classes $\mathfrak{C}$, where the class with the highest rate of positive outcomes is considered to be the *privileged class*, and the other classes are referred to as the *unprivileged classes*. If $\mathfrak{C}$ turns out to have cardinality >2, then the dataset or model being evaluated is considered to be *biased*.

Algorithm 1. Bias detection procedure.

procedure ComputePrivilegeClasses(D)
  $C = \emptyset$;
  $\mathfrak{C} = \emptyset$;
  $\mathfrak{C} = \emptyset$;
  $G \leftarrow$ Select $*$ From D GroupBy s;
  for all $g \in G$ **do**
    if $(len(C) \geq t)$ *and* $(disparate_{impact}(C, g) < 0.8)$ **then**
      $\mathfrak{C} \leftarrow \mathfrak{C} \cup \{C\}$;
      $C \leftarrow \emptyset$;
    end if
    $C \leftarrow C \cup g$;
  end for
  $\mathfrak{C} \leftarrow \mathfrak{C} \cup \{C\}$;
  return $\mathfrak{C}$;
end procedure

Following the procedure mentioned above, if the model results are unfair, the system provides the functionality to train an *unbiased version* of the same model. Algorithm 1 is used to split the model into two classes: the *privileged class*, i.e., the set of the sensitive feature values with the highest ratio of positive outcomes, and the *unprivileged class*, which contains the remaining values. It must be noted that there may be values of the sensitive feature for which no predictions are available yet. Based on the criterion described above, these values will be assigned to the unprivileged class. The rationale

behind this choice is that, in a situation where the system has no information about how a model perceives a specific value, assigning it to the unprivileged class, the system is guaranteed not to exacerbate pre-existing unknown prejudices.

Once the division of the feature values between privileged and unprivileged classes has been determined, the *reweighing algorithm* (Kamiran & Calders, 2012) is used for *bias mitigation*. This algorithm was chosen for multiple reasons: the system has access to the dataset used for training the inspected model, so a preprocessing strategy like the reweighing algorithm, which is likely to provide better results, can be applied; the reweighing algorithm bases its decision on the *independence* criterion, that is the exact fairness definition used to perform the distinction between privileged and unprivileged groups, and as previously mentioned, this definition has legal support; the algorithm output is a set of weights, which is easier to interpret than other techniques.

Once the new set of weights has been determined, the unbiased version of the inspected model can be trained using the same ML algorithm used for the original model and then stored and made available to be queried.

5. Case study

The focus of this section is on illustrating the application of the described system to the context of a loan approval process and providing an overview of the UI of the developed framework to show how it looks to users without further details about its technical implementation.

The diagram in Figure 6 shows how the functionalities explained in the previous section are accessible to the different types of users who can access the presented system through the developed framework.

Figure 7 shows the screen through which users can load a dataset. In our case study, the data has been provided to us by an Italian banking institution (after a *pseudonymization*[8] process) to build a prototype based on real data. The screen displays the characteristic of the loaded dataset and the results of preliminary bias detection.

After the dataset has been loaded users can select it and automatically train a new ML model (Figure 8) considering different algorithms (in the presented case study, since it is a binary problem, we have chosen *Logistic Regression*, *Random Forest*, and *Naive Bayes* algorithms). As the dataset is unbalanced, the system ranks the trained models based on the *F1-score* metric. It is also possible to compare the selected model with one of those already stored in the system.

After requesting a prediction, users can consult the model outcome with the relative *explanation* in an intuitive layout. As Figure 9 shows, the prediction result, along with its probability, is presented in the top-left box, while the related explanation is in the correct box (in the provided example,

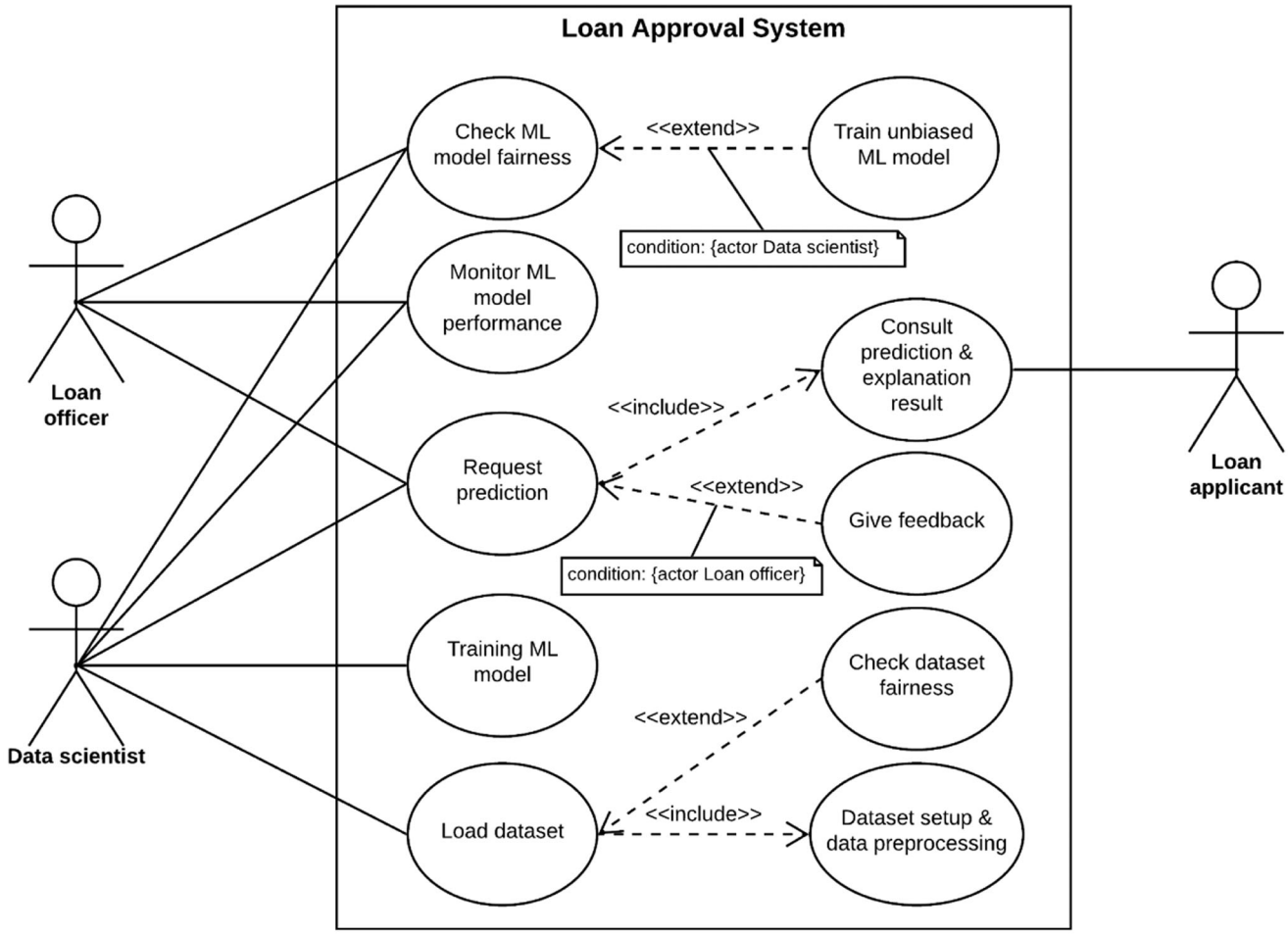

Figure 6. Use a case diagram of the presented system.

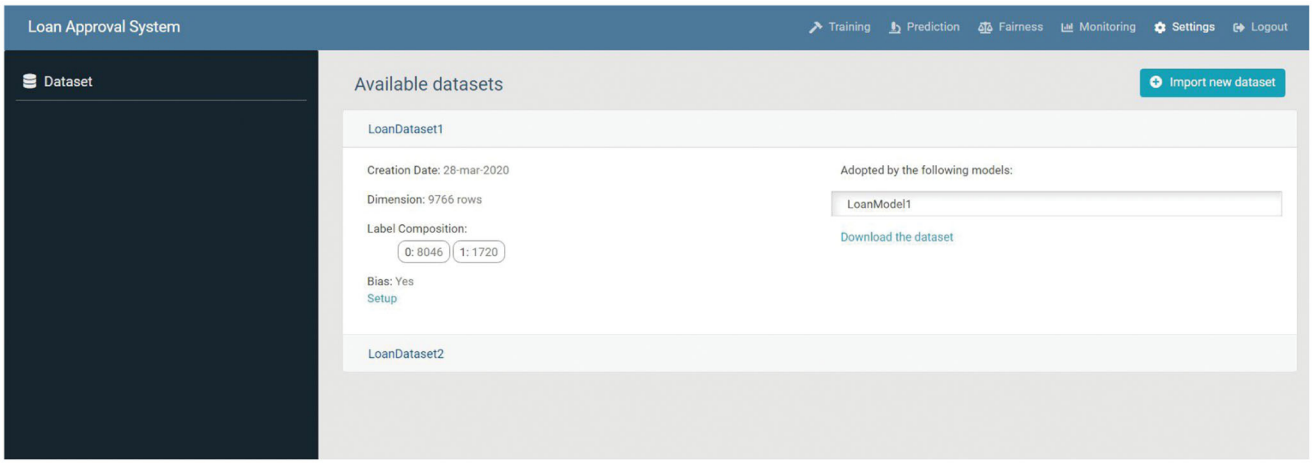

Figure 7. UI: load dataset.

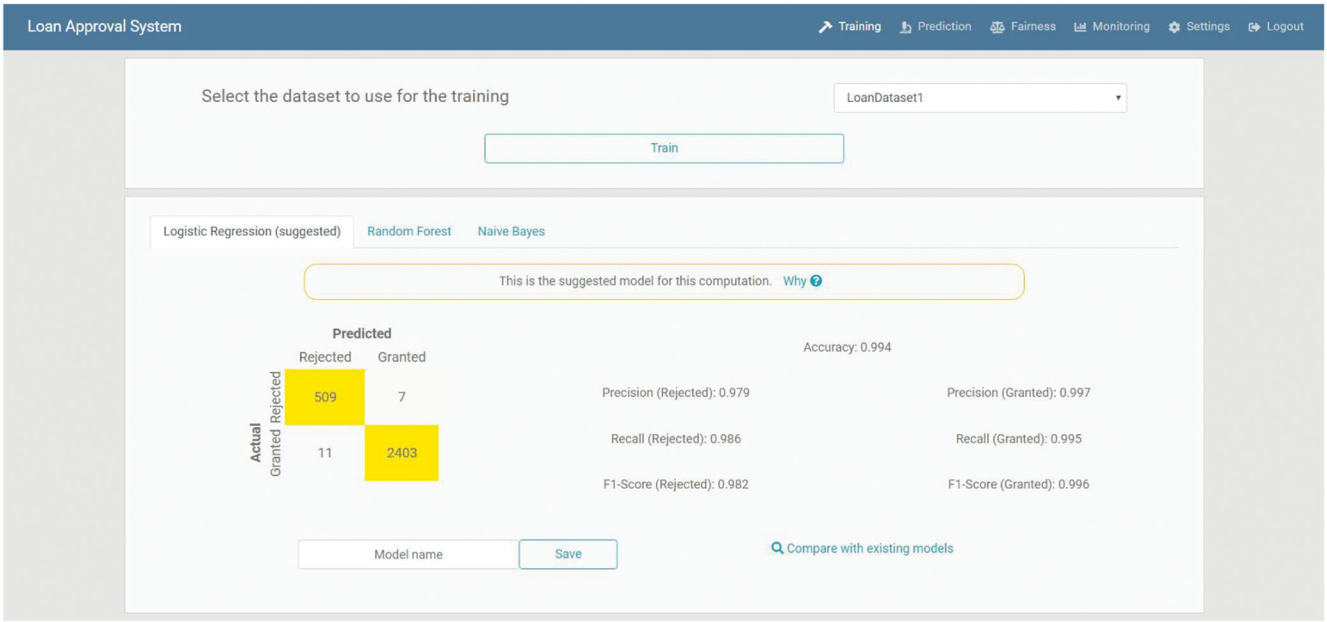

Figure 8. UI: ML model training.

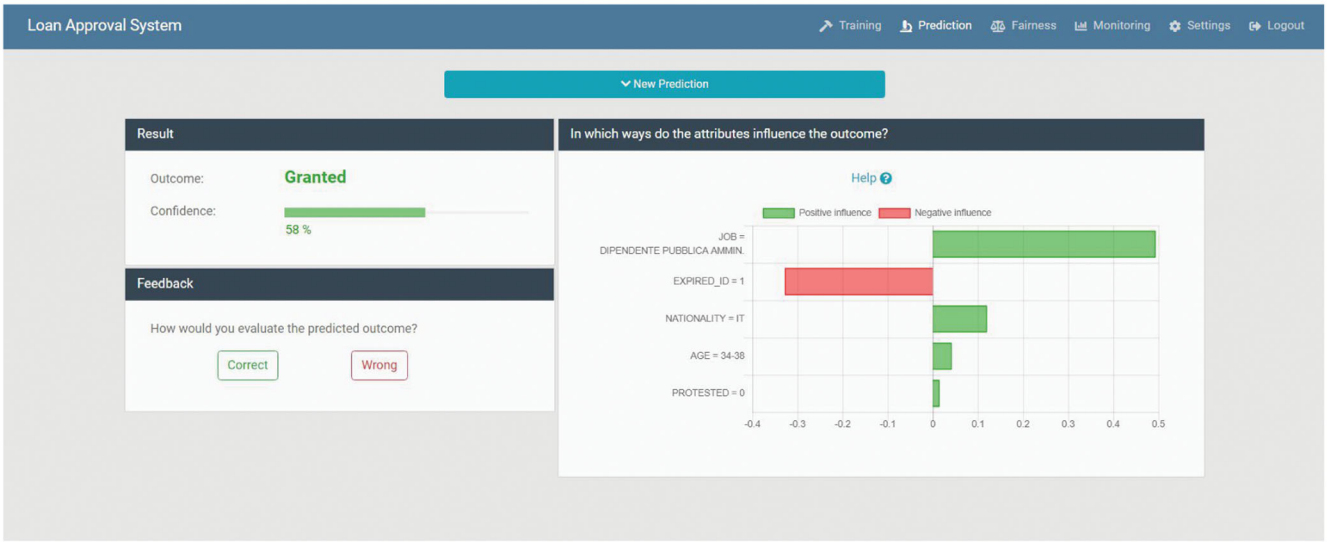

Figure 9. UI: displayed prediction and explanation output.

they are generated using the SHAP algorithm). Finally, the user can provide feedback on the prediction by clicking on the buttons contained in the bottom-left box.

By navigating to the "*Fairness*" tab, the user is presented with the privilege class division for the most recently uploaded dataset (Figure 10). In the presented case study, we have considered the "*nationality*" as the *sensitive feature*; the displayed partitioning is obtained by applying the procedure described in Algorithm 1. Within the same screen, selecting the "*Training*" option in the navigation menu on the left, users can request the system to train a new version of the model by exploiting the *reweighing* bias mitigation algorithm. Once the training is completed, the model is permanently stored in the system, along with its previous version, and can be selected in the "*Prediction*" tab of the main interface to request a prediction on it. In Figures 11 and 12 comparisons between the explanations generated by an unfair and a fair model for the same instance are shown. The navigation menu on the left side in Figure 10 provides the possibility to manually check the fairness of a dataset, although this functionality is automatically performed by the system when the dataset is loaded.

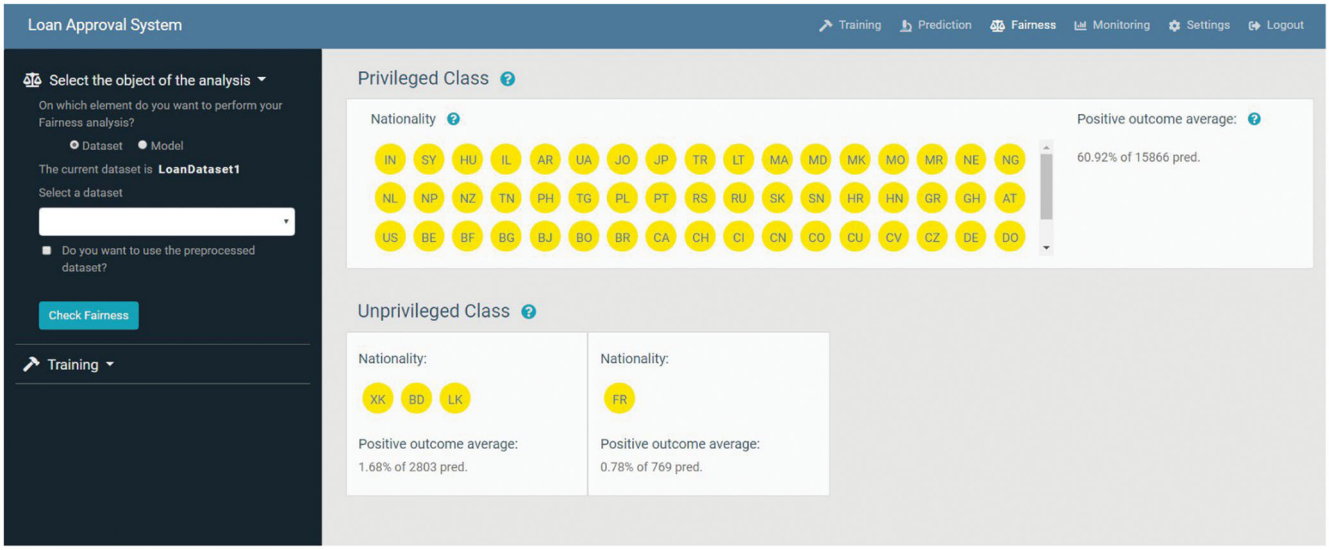

Figure 10. UI: bias detection.

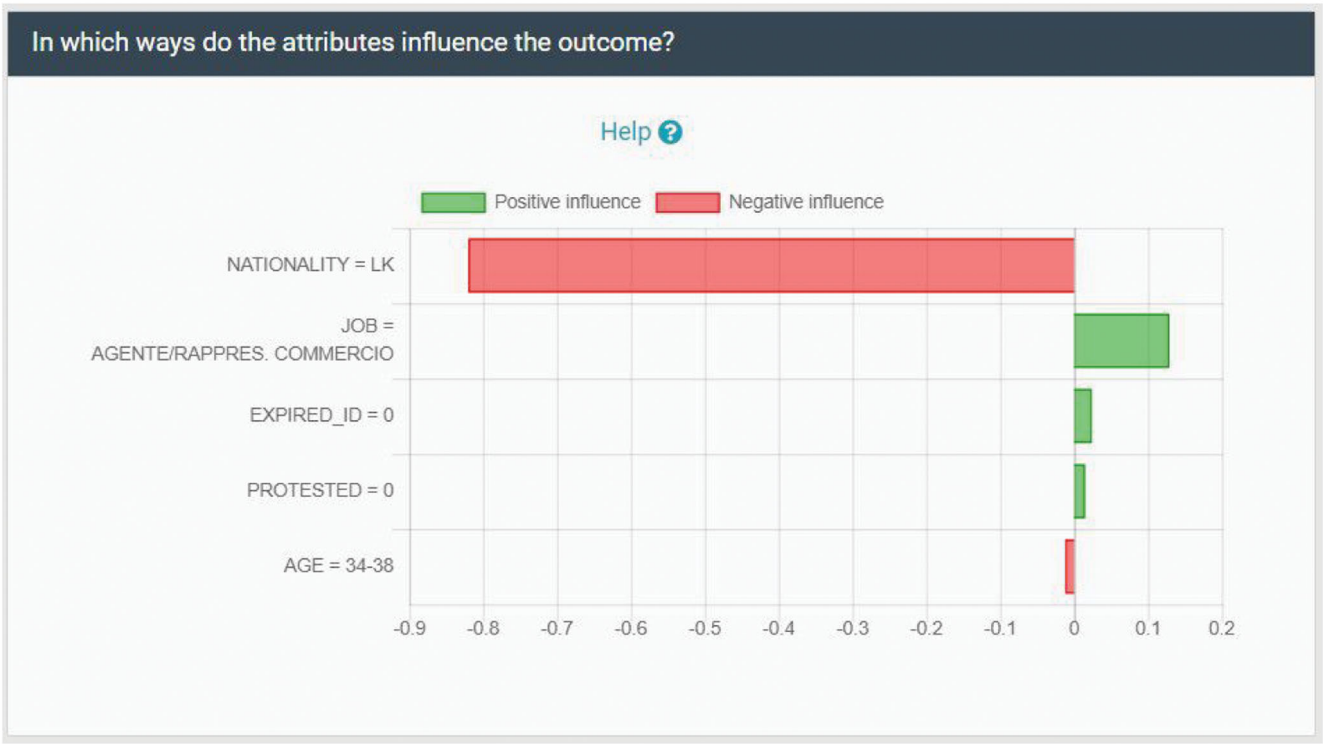

Figure 11. UI: Unfair model.

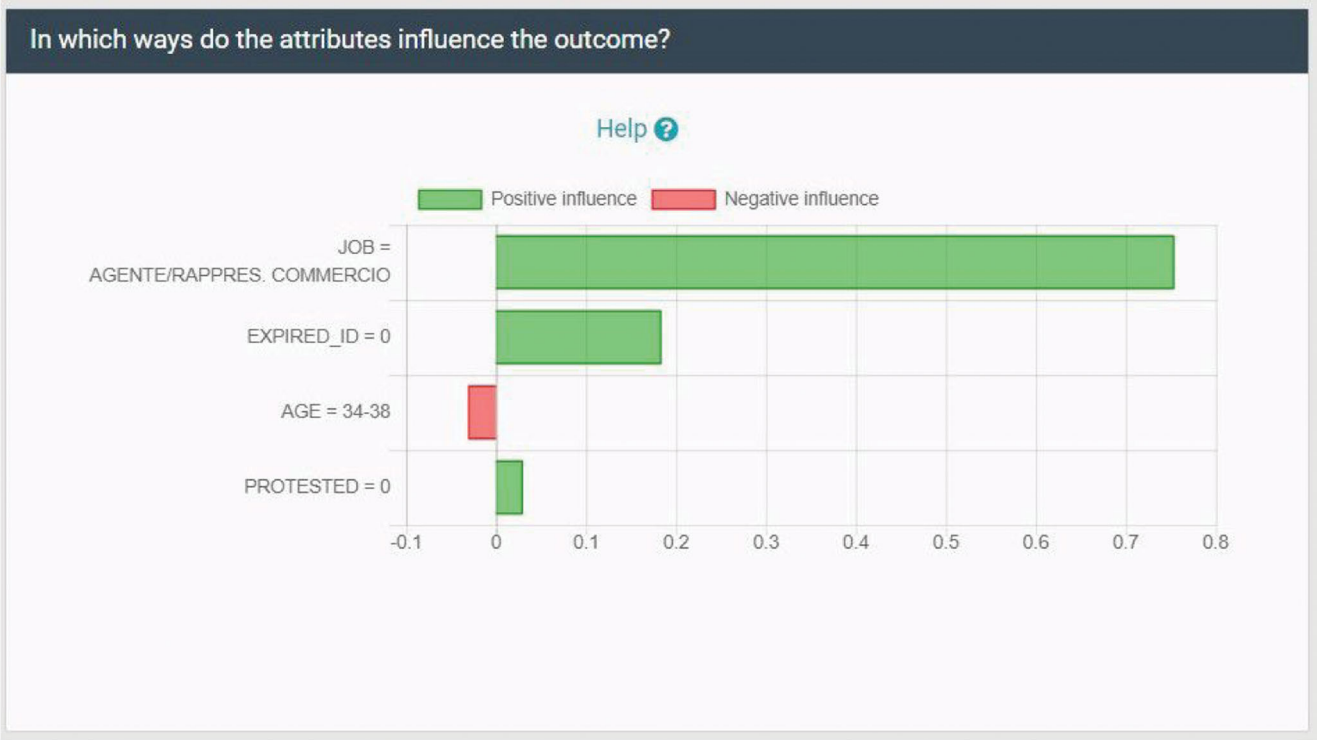

Figure 12. UI: Fair model.

6. Evaluation

The system presented in this article has been developed with the dual goal of supporting a loan approval process and laying out the foundation for a more general ML model management suite. This section is divided into four parts:

1. The first part shows the evaluation made for choosing the best explainability algorithm (among those usable in the developed system and described in Section 4.2) in the context of the presented case study; this evaluation has been carried out by a group of data scientists and researchers using the *Explanation Goodness Checklist* proposed by Hoffman et al. (2018);
2. The second part focuses on the experimental results of the *Loan Approval System* evaluation, from the explainability point of view, carried out through a novel *Trust & Reliance Scale* based on the *Trust Scale Recommended for XAI* proposed by Hoffman et al. (2018). Results are obtained by the submission of the mentioned novel scale to a group of bank domain experts and loan officers;
3. In the third part, the results of the *A/B test* and targeted interviews performed to evaluate the effectiveness of fairness usage in the presented system are displayed;
4. A *usability test* has been performed to measure user satisfaction with the UI, and the results are shown in the last part of this section.

6.1. Explainability algorithms evaluation

Hoffman et al. (2018) put together key concepts that have emerged from the literature in various fields of research (such as Philosophy of Science, Psychology, Education and Training, and Human Factors) and set guidelines for the evaluation of XAI systems.

To evaluate which explainability algorithm (LIME, SHAP, or Anchors) performs best in the context of the case study presented in Section 5, we exploit the *Explanation Goodness Checklist* proposed in the article mentioned above. This checklist represents a synopsis of the main features used in the research literature to consider explanations good. Quoting the authors, "*The intended use context is for researchers [...] to provide an independent, a priori evaluation of the goodness of explanations that are generated by [...] XAI systems.*" To be thorough, the checklist is reported in Appendix A.

For the experimental session, a pool of 54 people without experience with our framework have been asked to try the system through the developed UI for one month (between June and July 2020) and then compile the checklist. The group of participants was composed as follows: equally divided between data scientists and researchers in the computer science field, the majority (90%) are daily involved in ML model development; many (70%) are aware of XAI techniques. The division between males and females is 75–25%, and the average age is 27.2 years old. The pool has been split into three homogeneous subgroups (with respect to the factors mentioned earlier). We performed a *between-subject* evaluation: a different algorithm has been assigned to each of the three subgroups to make the evaluation process independent of influences due to having previously examined a different technique.

The three *post-hoc* methods have been applied to the same trained ML model with the following characteristics:

- Dataset size: 2440 samples;
- Training algorithm: Random Forest;

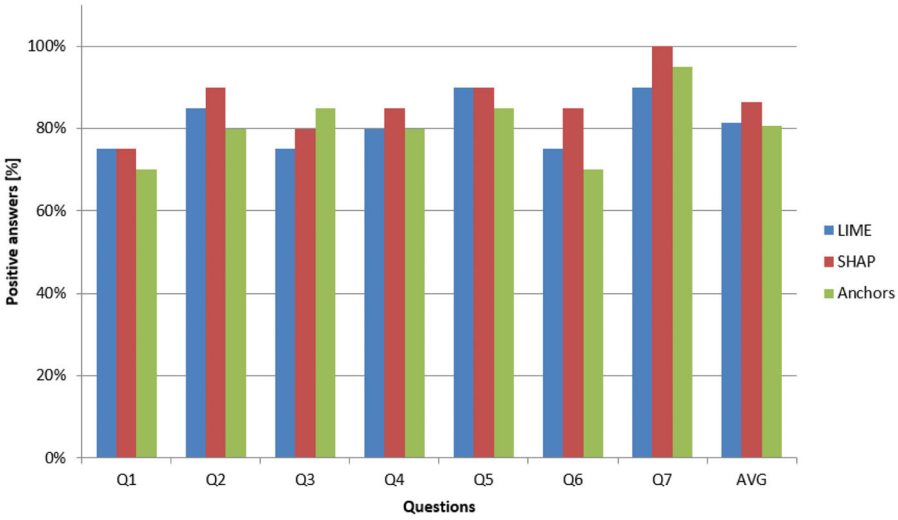

Figure 13. Explainability algorithm evaluation.

- Accuracy: 0.987;
- Precision/Recall (*Rejected*): 0.981/0.985;
- Precision/Recall (*Granted*): 0.997/0.996.

Figure 13 shows the percentage of positive answers (Y-axis) for each question (X-axis) and each algorithm. The evaluation outcomes display that *Anchors* are preferred only for the level of details provided, while *LIME*'s explanations are considered as understandable and actionable as *SHAP*'s, which overall result in the most satisfying, complete, accurate, reliable, and trustworthy. For this reason, we integrated this algorithm for the implementation of the system.

6.2. System evaluation (explainability)

A novel *Trust & Reliance Scale* is proposed in Appendix B. This scale, used to evaluate the effectiveness of predictions' explanations, is based mainly on the *Trust Scale Recommended for XAI* proposed by Hoffman et al. (2018) (Q1, Q3, Q4, Q6, and Q7). We have adapted the mentioned scale to achieve a new one that is better suited for the evaluation of our system according to the proposed approach. In particular, we have removed questions about predictability and efficiency and added three new items: a question derived from the work of Adams et al. (2003) (Q2) to ask users directly whether they trust the tool's output; a question from the Hoffman's *Explanation Satisfaction Scale* (Q8) for highlighting the judge about the importance of explanations; and another new question (Q5) to make users consider the possibility of trusting the system's response if it is different from theirs. This novel scale is realized as a *5-point Likert scale*, by following the literature, which indicates that the five-point format appears to be less confusing and tends to reduce the "frustration level" of respondents and thereby increase the response rate and the quality of the responses themselves (Babakus & Mangold, 1992; Devlin et al., 1993). For each of the corresponding statements, every user gives a response in the range between *Strongly disagree* and *Strongly agree*.

Since this kind of scale is addressed to users with considerable experience, it has been submitted to the group of participants after two months of continuous use of the system (October–November 2020). The group comprises 42 bank domain experts with experience in loan approval processes, 33 of whom are currently loan officers. All the practitioners belong to the Italian banking institution that provided the dataset to create the system prototype. The average age of the participants is 39.3 years old, and the average years of experience in loan approval processes are 9.6.

To build a baseline and be able to evaluate the actual improvement given by the explanations, we divided the pool of selected testers into two homogeneous subgroups and set up two different test environments: in the first one, the group was not aware of the explanations, and the UI has been modified to display the results of the predictions only with *label* and *confidence*, as shown in Figure 14. The second group, instead, interacted with the actual system prototype and the UI presented in Section 5 (see Figure 9).

The results of the two tests are shown, respectively, in Figures 15 and 16. We visualize the Likert scales with the *diverging stacked bar charts* as the graphical display technique, based on Robbins and Heiberger's studies on the presentation of results using rating scales (Heiberger & Robbins, 2014; Robbins & Heiberger, 2011).

By analyzing the results charts, it is clear how the possibility of checking the explanations of a given prediction has led to a better overall assessment of the system. Although in both test environments users had the perception that the system works adequately well (Q1) and most of them appreciate the use of such an automatic system to make these decisions (Q7), they have shown a concrete improvement in the system judgment in terms of trustworthiness and reliability (Q2, Q3, Q6). The explicit question about the usefulness of explanations in the second test (Q8) confirmed that perception. Due to displaying predictions' explanations, other noteworthy results are the overall decrease of "non-opinion" answers and the increase in the number of users that would change their minds based on the system's response (Q5). Finally, and perhaps surprisingly, in both environments, most users believe that such a

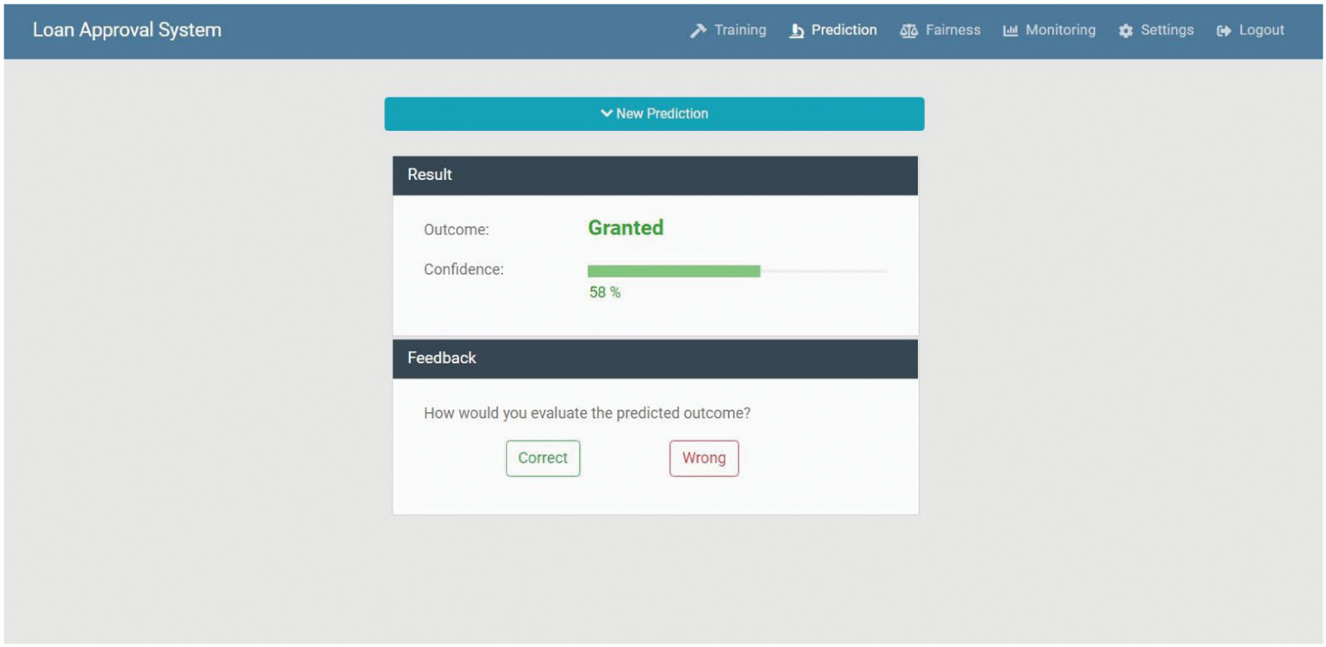

Figure 14. UI: displayed predictions *without* explanations.

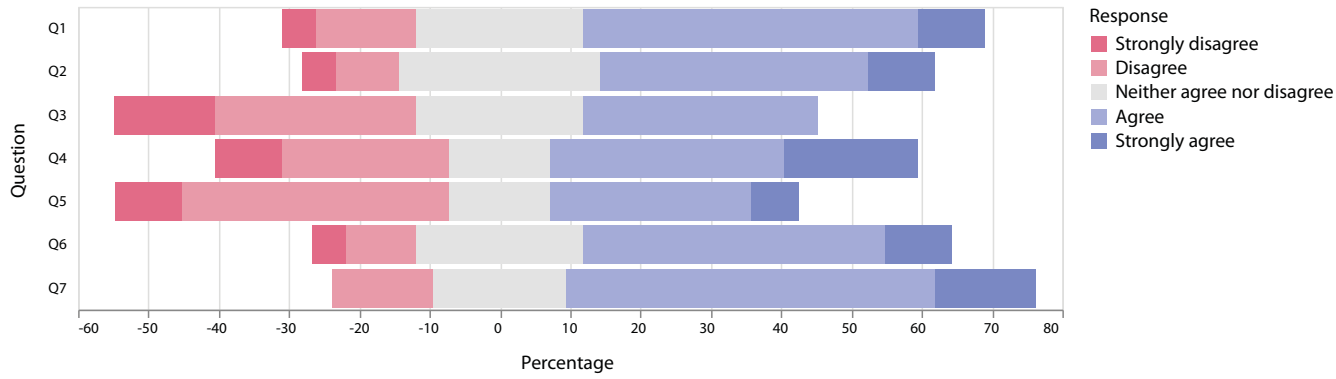

Figure 15. System evaluation *without* explanations (baseline).

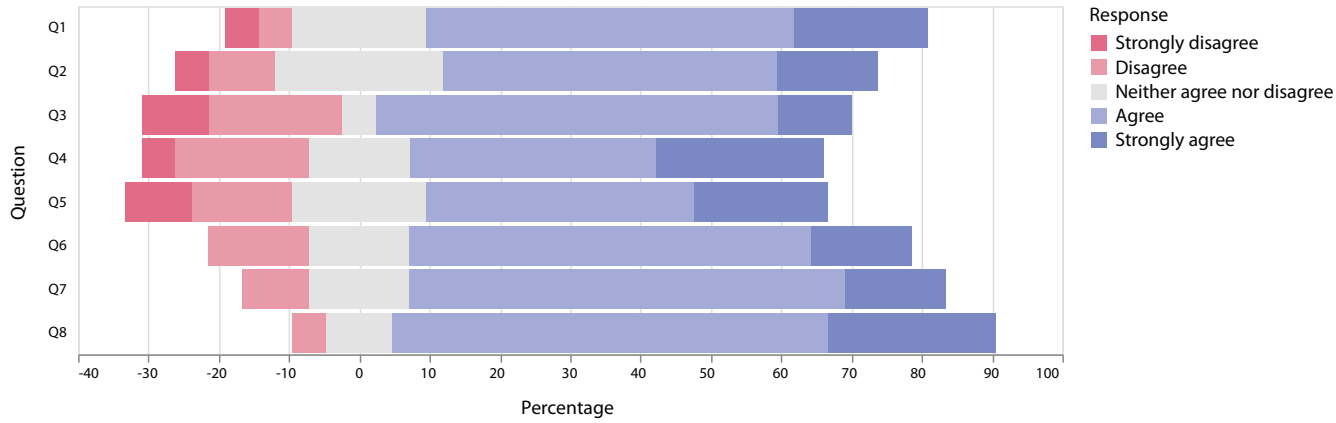

Figure 16. System evaluation *with* explanations.

system can give better results than a novice human (Q4). Moreover, we investigated the characteristics of the users who disagreed about reliability (Q3) and confidence (Q5) in the system with a displayed explanation. The resulting analysis showed that the average expertise in loan approval processes is 11.6 years, 2.1 years more than the overall average of the participants. This result underlines that experienced loan officers can be unenthusiastic about the use of new technologies in their daily work.

6.3. System evaluation (fairness)

To assess the effectiveness of the fairness, we decided to set up an *A/B test* as described in the following. We first selected

24 loan officers (average age of 34.6 years old and average experience in the field of 5.5 years) not involved in the previous evaluation to participate in a test session for checking the usefulness of the feedback loop. For each displayed prediction and related explanations, their task was to judge whether it was correct by clicking on the specific button on the bottom-left part of the UI shown in Figure 9. The participants, however, were unconsciously divided into two homogeneous subgroups to assess the possible differences in the evaluation of predictions correctness based on the use of 2 opposed models. The first group interacted with an unfair model, as in Figure 11, while the second one with a fair model, as in Figure 12, where the *nationality* attribute was missing at all.

The assessment, lasting 2 hr, was carried out by showing each user 50 predictions. Figure 17 displays the results in terms of click rate on the feedback buttons.

As can be seen from the chart, the percentage of negative responses is higher for the unfair-model testers. Based on this outcome, in the second part of the evaluation, we performed a series of targeted interviews with some of the loan officers involved in both interactions. The most relevant conclusion is that, while the 92% of the "fair-model testers" stated that they "*focused on assessing the actual correctness of the prediction based on their experience*," the 88% of the "unfair-model testers" confirmed that often their attention was just on the nationality attribute weight because they would "*never agree to confirm the rejection or the approval of a loan application in which the greater weight of the decision is attributable to a potential discriminatory individual characteristic such as the nationality of the applicant*." Although not part of that test, they all agreed to consider the visualization of the predictions' explanations as an essential feature for this kind of automated system.

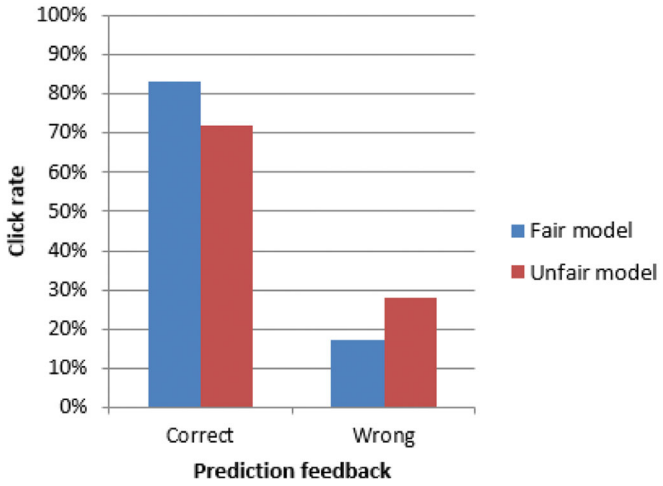

Figure 17. Fairness evaluation.

6.4. User interface evaluation

Finally, we present a qualitative evaluation of the developed UI to measure user satisfaction with the system usability.

This experimental session was attended by the bank domain experts already involved in the previous system evaluation. The questionnaire, reported in Appendix C, is based on the usability test proposed by Purificato and Rinaldi (2018) and structured following a methodology presented by IBM (Lewis, 1995), but adapted to a five-point format for the abovementioned motivations. Each participant tested the three functionalities for one month (March 2021) and then evaluated them with the same procedure as the one described in the previous section.

We had the three main functionalities of the system tested (dataset and ML model handler, explainability Tool, and fairness Tool) and the results displayed, respectively, in Figures 18–20, allow us to state that users consider the developed UI effective. Some improvement is required for

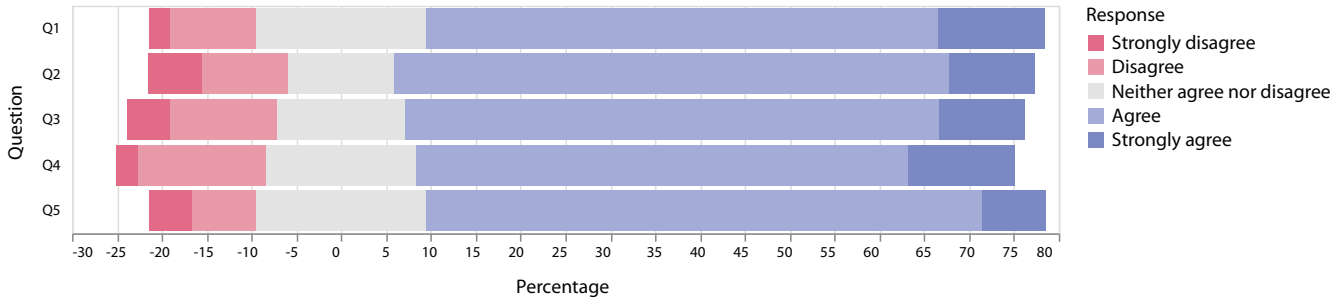

Figure 18. UI evaluation results: dataset and ML model handler.

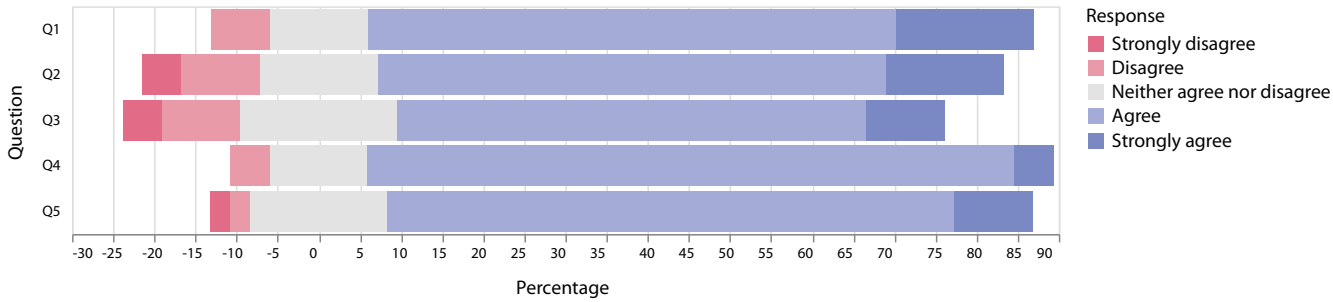

Figure 19. UI evaluation results: explainability tool.

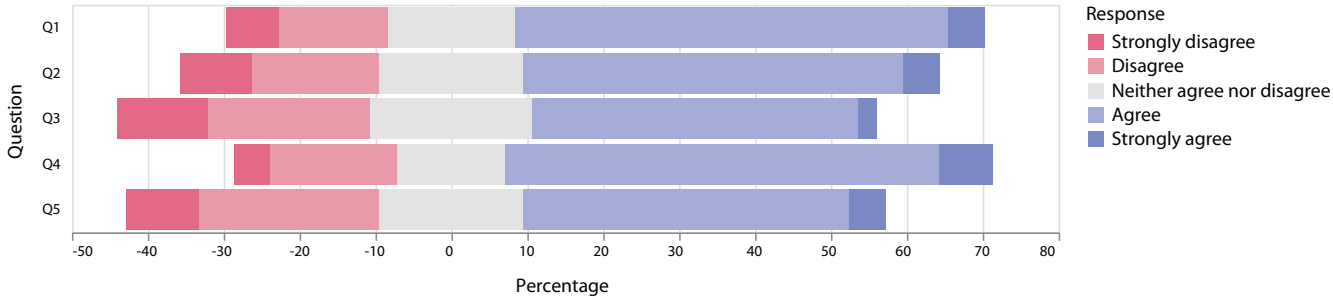

Figure 20. UI evaluation results: fairness tool.

the fairness tool in terms of complexity in finding the needed information to carry out the specific task.

7. Conclusion and discussion

In this article, we presented a system that focuses on two of the fundamental ethical principles in the fields of Responsible and Trustworthy AI, *explainability* and *fairness*. The system is applied to the context of loan approval processes through the implementation of a proprietary framework able to manage the whole life cycle of an ML model to show how the use of explainability and fairness techniques can lead to the growth of a bank domain expert's *trust* and *reliance* on AI systems.

With this aim, four functionalities have been designed and developed: a *dataset & ML model handler*, a *standardized explainability tool*, a *fairness tool*, and a *Feedback loop*. In particular, the standardized Explainability tool provides methods to get explanations for each prediction, allowing users to choose among three different algorithms: *LIME*, *SHAP*, and *Anchors*. The Fairness tool allows users to detect biases within the model's behavior through a proposed algorithm based on *disparate impact* metrics, and to mitigate them using the *reweighing* algorithm, a method following the *independence* criterion, one of the few criteria that have legal support. An *unbiased* version of the original model can be trained at the end of the described procedure.

A proprietary framework with an attractive and easy-to-use UI has been developed, and the whole system has been evaluated in the context of loan approval processes.

The effectiveness of our approach has been proven through experimental results from field tests and user studies. *SHAP* has been chosen as the preferred explainability algorithm through the submission of the *Explanation Goodness Scale* to a group composed of data scientists and researchers. The enhanced trust in the use of our system has been assessed by bank domain experts through a novel *Trust & Reliance Scale* proposed in the article. Finally, a *Usability Test* has demonstrated the usefulness of the developed user interface.

Notes

1. https://2021.ai/fairness-in-machine-learning/. Last seen May 24, 2022.
2. "*Correctional Offender Management Profiling for Alternative Sanctions*" is a proprietary algorithm sold by *Equivant*, a private company founded in 2017 as a rebranding of Northpointe, Inc., CourtView Justice Solutions, Inc., and Constellation Justice Systems, Inc.
3. IBM Watson OpenScale "Manage AI, with trust and confidence in business outcomes". White paper. https://www.ibm.com/downloads/cas/RYXBG8OZ. Last seen May 24, 2022.
4. Application programming interface.
5. Google. "What If … you could inspect a machine learning model, with minimal coding required?". https://pair-code.github.io/what-if-tool/index.html. Last seen May 24, 2022.
6. Google. "AI Explainability Whitepaper". https://storage.googleapis.com/cloud-ai-whitepapers/AI%20Explainability%20Whitepaper.pdf. As of today, May 24, 2022, it is in *beta* version.
7. First release in late September 2019.
8. It means the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organizational measures to ensure that the personal data are not attributed to an identified or identifiable natural person.

Acknowledgments

The authors want to thank Blue Reply S.p.A for supporting the realization of this work, developed in its early stages as Flavio Lorenzo's Master Thesis at the Polytechnic University of Turin, between July and December 2019.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Erasmo Purificato http://orcid.org/0000-0002-5506-3020
Francesca Fallucchi http://orcid.org/0000-0002-3288-044X
Ernesto William De Luca http://orcid.org/0000-0003-3621-4118

References

Abdul, A., Vermeulen, J., Wang, D., Lim, B. Y., & Kankanhalli, M. (2018). *Trends and trajectories for explainable, accountable and intelligible systems: An HCI research agenda* [Paper presentation]. Proc. of the 2018 CHI Conference on Human Factors in Computing Systems (pp. 1–18). https://doi.org/10.1145/3173574.3174156

Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access*, *6*, 52138–52160. https://doi.org/10.1109/ACCESS.2018.2870052

Adams, B. D., Bruyn, L. E., & Houde, S. (2003). *Trust in automated systems, literature review*. Humansystems Incorporated.

AI-HLEG (2019). *Ethics guidelines for trustworthy AI*.

Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016, May). *Machine bias – There's software used across the country to predict future criminals. And it's biased against blacks* (Technical report). ProPublica.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Arun, K., Ishan, G., & Sanmeet, K. (2016). Loan approval prediction based on machine learning approach. *IOSR Journal of Computer Engineering*, *18*(3), 18–21.

Arya, V., Bellamy, R. K., Chen, P.-Y., Dhurandhar, A., Hind, M., & Hoffman, S. C. (2019). One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. *arXiv* 1909.03012.

Babakus, E., & Mangold, W. G. (1992). Adapting the servqual scale to hospital services: An empirical investigation. *Health Services Research*, *26*(6), 767–786.

Bellamy, R. K., Dey, K., Hind, M., Hoffman, S. C., Houde, S., & Kannan, K. (2018). AI fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv* 1810.01943.

Biddle, D. (2006). *Adverse impact and test validation: A practitioner's guide to valid and defensible employment testing*. Gower Publishing, Ltd.

Binns, R. (2020). *On the apparent conflict between individual and group fairness* [Paper presentation]. Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (pp. 514–524).

Biran, O., & Cotton, C. (2017). Explanation and justification in machine learning: A survey. In *IJCAI-17 workshop on explainable AI (XAI)* (Vol. 8). AAAI Press.

Board, F. S. (2017). *Artificial Intelligence and machine learning in financial services: Market developments and financial stability implications, 1 November 2017*.

Calmon, F., Wei, D., Vinzamuri, B., Ramamurthy, K. N., & Varshney, K. R. (2017). Optimized pre-processing for discrimination prevention. In *Advances in neural information processing systems* (pp. 3992–4001). Curran Associates, Inc.

Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, *5*(2), 153–163. https://doi.org/10.1089/big.2016.0047

Commission, E. (n.d.). *Can I be subject to automated individual decision-making, including profiling?* FAQ. Retrieved from https://ec.europa.eu/info/law/law-topic/data-protection/reform/rights-citizens/my-rights/can-i-be-subject-automated-individual-decision-making-including-profiling_en

Corbett-Davies, S., & Goel, S. (2018). The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv* 1808.00023.

DataRobot (2019). *Intelligence briefing: How banks are winning with AI and automated machine learning* (White Paper). DataRobot.

Devlin, S. J., Dong, H., & Brown, M. (1993). Selecting a scale for measuring quality. *Marketing Research*, *5*(3), 12–17.

Dhurandhar, A., Pedapati, T., Balakrishnan, A., Chen, P.-Y., Shanmugam, K., & Puri, R. (2019). Model agnostic contrastive explanations for structured data. *arXiv* 1906.00117.

Dignum, V. (2018). *Ethics in artificial intelligence: Introduction to the special issue*. Springer. https://doi.org/10.1007/s10676-018-9450-z

Dignum, V. (2019). *Responsible artificial intelligence: How to develop and use AI in a responsible way*. Springer Nature.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). *Fairness through awareness* [Paper presentation]. Proc. of the 3rd Innovations in Theoretical Computer Science Conference (pp. 214–226). https://doi.org/10.1145/2090236.2090255

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., & Venkatasubramanian, S. (2015). *Certifying and removing disparate impact* [Paper presentation]. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 259–268).

Friedler, S. A., Scheidegger, C., & Venkatasubramanian, S. (2016). On the (im)possibility of fairness. *arXiv* 1609.07236.

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). *Explaining explanations: An overview of interpretability of machine learning* [Paper presentation]. 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 80–89). https://doi.org/10.1109/DSAA.2018.00018

Goebel, R., Chander, A., Holzinger, K., Lecue, F., Akata, Z., Stumpf, S., Kieseberg, P., & Holzinger, A. (2018). *Explainable ai: the new 42?* [Paper presentation]. International Cross-Domain Conference for Machine Learning and Knowledge Extraction (pp. 295–303).

Goodman, B., & Flaxman, S. (2017). European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, *38*(3), 50–57. https://doi.org/10.1609/aimag.v38i3.2741

Gunning, D. (2017). *Explainable artificial intelligence (XAI)*. Defense Advanced Research Projects Agency (DARPA).

Hagenau, M., Liebmann, M., & Neumann, D. (2013). Automated news reading: Stock price prediction based on financial news using context-capturing features. *Decision Support Systems*, *55*(3), 685–697. https://doi.org/10.1016/j.dss.2013.02.006

Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. In *Advances in neural information processing systems* (pp. 3315–3323). MIT Press.

Heaton, J., Polson, N. G., & Witte, J. H. (2016). Deep learning in finance. *arXiv* 1602.06561.

Heiberger, R. M., & Robbins, N. B. (2014). Design of diverging stacked bar charts for Likert scales and other applications. *Journal of Statistical Software*, *57*(5), 1–32. https://doi.org/10.18637/jss.v057.i05

Hirasawa, T., Aoyama, K., Tanimoto, T., Ishihara, S., Shichijo, S., Ozawa, T., Ohnishi, T., Fujishiro, M., Matsuo, K., Fujisaki, J., & Tada, T. (2018). Application of artificial intelligence using a convolutional neural network for detecting gastric cancer in endoscopic images. *Gastric Cancer*, *21*(4), 653–660. https://doi.org/10.1007/s10120-018-0793-2

Hoffman, R. R., Mueller, S. T., Klein, G., & Litman, J. (2018). Metrics for explainable AI: Challenges and prospects. *arXiv* 1812.04608.

Holstein, K., Wortman Vaughan, J., Daumé, H. III, Dudik, M., & Wallach, H. (2019). Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (pp. 1–16).

Holzinger, A., Biemann, C., Pattichis, C. S., & Kell, D. B. (2017). What do we need to build explainable ai systems for the medical domain? *arXiv* 1712.09923.

Jarrahi, M. H. (2018). Artificial intelligence and the future of work: Human-AI symbiosis in organizational decision making. *Business Horizons*, *61*(4), 577–586. https://doi.org/10.1016/j.bushor.2018.03.007

Kamiran, F., & Calders, T. (2012). Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, *33*(1), 1–33. https://doi.org/10.1007/s10115-011-0463-8

Kamiran, F., Karim, A., & Zhang, X. (2012). Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining* (pp. 924–929).

Kapishnikov, A., Bolukbasi, T., Viégas, F., & Terry, M. (2019). Xrai: Better attributions through regions. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 4948–4957).

Kleinberg, J., Mullainathan, S., & Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores. *arXiv* 1609.05807.

Lamy, J.-B., Sekar, B., Guezennec, G., Bouaud, J., & Séroussi, B. (2019). Explainable artificial intelligence for breast cancer: A visual case-based reasoning approach. *Artificial Intelligence in Medicine*, *94*, 42–53. https://doi.org/10.1016/j.artmed.2019.01.001

Landecker, W., Thomure, M. D., Bettencourt, L. M., Mitchell, M., Kenyon, G. T., & Brumby, S. P. (2013). *Interpreting individual classifications of hierarchical networks* [Paper presentation]. 2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM) (pp. 32–38). https://doi.org/10.1109/CIDM.2013.6597214

Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, *7*(1), 57–78. https://doi.org/10.1080/10447319509526110

Liu, Y., Radanovic, G., Dimitrakakis, C., Mandal, D., & Parkes, D. C. (2017). Calibrated fairness in bandits. *arXiv* 1707.01875.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in neural information processing systems* (pp. 4765–4774). MIT Press.

Maleki, S., Tran-Thanh, L., Hines, G., Rahwan, T., & Rogers, A. (2013). Bounding the estimation error of sampling-based Shapley value approximation. *arXiv* 1306.4265.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2019). A survey on bias and fairness in machine learning. *arXiv* 1908.09635.

Millecamp, M., Htun, N. N., Conati, C., & Verbert, K. (2019). *To explain or not to explain: the effects of personal characteristics when explaining music recommendations* [Paper presentation]. Proceedings of the 24th International Conference on Intelligent User Interfaces (pp. 397–407).

Miller, T. (2019a). "But why?" Understanding explainable artificial intelligence. *XRDS: Crossroads, the ACM Magazine for Students*, *25*(3), 20–25. https://doi.org/10.1145/3313107

Miller, T. (2019b). Explainable artificial intelligence: What were you thinking? In *Artificial intelligence for better or worse*.

Miller, T. (2019c). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, *267*, 1–38. https://doi.org/10.1016/j.artint.2018.07.007

Mueller, S. T., Hoffman, R. R., Clancey, W., Emrey, A., & Klein, G. (2019). Explanation in human-ai systems: A literature meta-review, synopsis of key ideas and publications, and bibliography for explainable ai. *arXiv* 1902.01876.

Palczewska, A., Palczewski, J., Robinson, R. M., & Neagu, D. (2014). Interpreting random forest classification models using a feature contribution method. In *Integration of reusable systems* (pp. 193–218). Springer.

Preece, A., Harborne, D., Braines, D., Tomsett, R., & Chakraborty, S. (2018). Stakeholders in explainable ai. *arXiv* 1810.00184.

Purificato, E., & Rinaldi, A. M. (2018). A multimodal approach for cultural heritage information retrieval. In *International Conference on Computational Science and Its Applications* (pp. 214–230).

Rajani, N. F., & Mooney, R. J. (2018). Ensembling visual explanations. In *Explainable and interpretable models in computer vision and machine learning* (pp. 155–172). Springer.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016a). Nothing else matters: Model-agnostic explanations by identifying prediction invariance. *arXiv* 1611.05817.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016b). "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144).

Ribeiro, M. T., Singh, S., & Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Robbins, N. B., & Heiberger, R. M. (2011). Plotting likert and other rating scales. In *Proceedings of the 2011 Joint Statistical Meeting* (pp. 1058–1066).

Saxena, N. A., Huang, K., DeFilippis, E., Radanovic, G., Parkes, D. C., & Liu, Y. (2020). How do fairness definitions fare? Testing public attitudes towards three algorithmic definitions of fairness in loan allocations. *Artificial Intelligence*, *283*, 103238. https://doi.org/10.1016/j.artint.2020.103238

Schumaker, R. P., & Chen, H. (2010). A discrete stock price prediction engine based on financial news. *Computer*, *43*(1), 51–56. https://doi.org/10.1109/MC.2010.2

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 618–626).

Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, *2*(28), 307–317. http://dx.doi.org/10.1515/9781400881970-018

Shrikumar, A., Greenside, P., Shcherbina, A., & Kundaje, A. (2016). Not just a black box: Learning important features through propagating activation differences. *arXiv* 1605.01713.

Simonite, T. (2018, December). *Google's AI Guru wants computers to think more like brains* (Technical Report). Wired.

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning* (Vol. 70, pp. 3319–3328).

Tamagnini, P., Krause, J., Dasgupta, A., & Bertini, E. (2017). *Interpreting black-box classifiers using instance-level visual explanations* [Paper presentation]. Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics (pp. 1–6).

Wang, C., Wang, K., Bian, A., Islam, R., Keya, K. N., Foulds, J., & Pan, S. (2022). *Do humans prefer debiased ai algorithms? a case study in career recommendation* [Paper presentation]. 27th International Conference on Intelligent User Interfaces (pp. 134–147). https://doi.org/10.1145/3490099.3511108

Washington, A. L. (2019). How to argue with an algorithm: Lessons from the compas-propublica debate. *The Colorado Technology Law Journal*, *17*(1), 131.

Yeom, S., & Tschantz, M. C. (2018). Discriminative but not discriminatory: A comparison of fairness definitions under different worldviews. *arXiv* 1808.08619.

Zafar, M. B., Valera, I., Gomez Rodriguez, M., & Gummadi, K. P. (2017). *Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment* [Paper presentation]. Proceedings of the 26th International Conference on World Wide Web (pp. 1171–1180).

Zhang, B. H., Lemoine, B., & Mitchell, M. (2018). *Mitigating unwanted biases with adversarial learning* [Paper presentation]. Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society (pp. 335–340).

Zhu, J., Liapis, A., Risi, S., Bidarra, R., & Youngblood, G. M. (2018). *Explainable ai for designers: A human-centered perspective on mixed-initiative co-creation* [Paper presentation]. 2018 IEEE Conference on Computational Intelligence and Games (CIG) (pp. 1–8). https://doi.org/10.1109/CIG.2018.8490433

About the Authors

Erasmo Purificato is a PhD student at the Otto von Guericke University Magdeburg. His research topics focus on user profiling, human-computer interaction and responsible AI. He is also a research fellow at the Leibniz Institute for Educational Media, Georg Eckert Institute, where he is coordinating the Usability Lab.

Flavio Lorenzo is a software engineer and data scientist from Brescia, Italy. He obtained his Master degree in computer engineering at the Polytechnic University of Turin in 2019 with a thesis on XAI. Formerly a consultant at Blue Reply, he's currently driving the IT modernization of his family-owned company, Cafcom.

Francesca Fallucchi is a researcher of the University of Rome Guglielmo Marconi and she is an information scientist at the Leibniz Institute for Educational Media, Georg Eckert Institute. Her main interests cover knowledge organization, information retrieval, semantic technology, big data and digital humanities.

Ernesto William De Luca is head of the "Digital Information and Research Infrastructures" department at the Georg Eckert Institute and Full Professor in "Research Infrastructures for Digital Humanities" at the Otto von Guericke University Magdeburg. Furthermore, he is a professor in computational engineering at the University of Rome Guglielmo Marconi.

Appendix A. Explanation goodness checklist

1. The explanation helps me understand how the tool works.
   YES □ **NO** □
2. The explanation of how the tool works is satisfying.
   YES □ **NO** □
3. The explanation of the tool sufficiently detailed.
   YES □ **NO** □
4. The explanation of how the tool works is sufficiently complete.
   YES □ **NO** □
5. The explanation is actionable, i.e., it helps me know how to use the tool.
   YES □ **NO** □
6. The explanation lets me know how accurate or reliable the algorithm is.
   YES □ **NO** □
7. The explanation lets me know how trustworthy the tool is.
   YES □ **NO** □

Appendix B. Trust & reliance scale

(1) I am **confident** in the tool. I feel it works well.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(2) I **trust** the tool's output.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(3) The tool is **reliable**. I can count on it to be correct all the time.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(4) The tool can **perform** the task **better** than a novice human user.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(5) If need be, I feel **confident** in considering changing my decision by taking the tool's output.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(6) I feel **safe** that when I rely on the tool I will get the right answer.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(7) I **like** using the tool for decision making.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(8) The explanations let me judge when I should **trust** and **not trust** the tool.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

Appendix C. Usability test

(1) Overall, I am satisfied with how easy it is to use this system.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(2) It was simple to use this system.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(3) I was able to complete the tasks quickly using this system.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(4) It was easy to learn to use this system.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree

(5) It was easy to find the information I needed.

Strongly disagree Disagree Neither agree nor disagree Agree Strongly agree