Investigating Usability Indicators for the Adoption of AI Models in Heuristic Evaluation

Demerval Gomes S. Júnior[1](✉), Rodrigo Hernández-Ramírez[1], and Jacinto Estima[2]

[1] UNIDCOM/IADE – Unidade de Investigação em Design e Comunicação, Av. D. Carlos I, 4, 1200-649 Lisbon, Portugal
dgsjunior@gmail.com, rodrigo.ramirez@universidadeeuropeia.pt
[2] Department of Informatics Engineering, CISUC/University of Coimbra, R. Silvio Lima, 3030-790 Coimbra, Portugal
estima@dei.uc.pt

Abstract. Online Learning Management Systems (LMS) have become widely used solutions over the last few years by educational institutions worldwide. Interest in evaluating the quality of these systems has been increasing, and new research to investigate the usability and user experience (UX) of these platforms has increased over the last decade. One of the common evaluation approaches is the heuristic evaluation of the interface based on selected criteria or indicators that describe well-known usability problems. However, this process remains laborious and challenging, requiring considerable effort from evaluators. Adopting automated methods is still uncommon, and approaches based on Artificial Intelligence (AI), for example, are rare. This article presents a study that investigates the potential adoption of usability indicators (Ui) for using artificial intelligence methods as supportive tools for heuristic evaluation of LMS interfaces. In our study, we developed a methodology to investigate some requirements to identify and select a set of Ui to create datasets for AI models to contribute to LMS interface inspection. The methodology allowed us to highlight a set of Ui to be potentially adopted with Machine Learning (ML) to evaluate LMS interfaces. We highlight a set of necessary assumptions to build datasets that can be used with AI models for heuristic evaluation. The methodological approach we propose can be repurposed to study new usability indicators to analyze other complex software contexts.

Keywords: Heuristic Evaluation · Usability and UX · Usability Indicators · Learning Management System · Artificial Intelligence

1 Introduction

Distance Learning (DL) is commonly supported by a type of system known as an Online Learning Environment (OLE), primarily represented by Learning Management Systems (LMS). These platforms are developed to contribute to the teaching and learning process through effective and efficient interaction between educators and students [1]. LMS

N. Martins and D. Brandão (Eds.): DIGICOM 2023, SSDI 35, pp. 136–151, 2024.
https://doi.org/10.1007/978-3-031-47281-7_11

encompass functionalities of content management and control, management of pedagogical activities, and communications within the educational context. The adoption of LMS can be either full or partial. E-learning encompasses two primary modalities: fully online courses or a blended learning model. In the first format, students and educators are not physically present in the same location, whereas in the latter case, a portion of the activities takes place in person, while another part occurs in an online environment [2]. Generally, LMS support synchronous and asynchronous learning processes. In the synchronous format, students and educators are online simultaneously, engaging in interactive communication within the same timeframe. In the asynchronous format, interaction occurs intermittently and, therefore, not necessarily at the same time. Communication between educators and students in LMS takes place through discussion groups mediated by the use of tools such as forums, email, chats, and other integrated solutions within the system. Nowadays, the most widely used LMS belong to two main categories: open-source and closed-source. Open-source solutions are based on free access to source code under certain conditions. Some examples include *Moodle*, *Sakai*, and *dotLRN* [1, 3]. Closed-source options are proprietary and do not provide access to the source code for consultation or reuse. Some examples are *Blackboard*, *Ping Pong*, *Canvas LMS*, *McGraw-Hill Education*, *D2L*, *and Blackbaud* [1, 3].

In recent years, LMS have attained a significant role in educational institutions worldwide, particularly after the COVID-19 Pandemic. Furthermore, new technological possibilities have contributed to the adoption of these systems. However, it is not uncommon to see objections and dissatisfaction from students and educators regarding the quality of these platforms, particularly in terms of usability and User Experience (UX) [1, 3–5, 5, 6]. In this context, ensuring continuous improvement of LMS through frequent evaluation of their interfaces becomes imperative. However, these evaluations are usually conducted from the perspective of usability and UX [1, 3, 6].

Regarding usability, the practical aspects of the daily activities of educators and students are evaluated [1, 3, 6]. Regarding UX, subjective matters are examined, particularly concerning hedonistic aspects related to the emotional responses that the platforms elicit [7]. When an LMS is introduced, educators and students will need time to adapt to the system and become proficient in performing typical tasks satisfactorily. This adaptation process can be more challenging if the interface presents usability issues, as users will need to overcome the inevitable frictions. Although evaluating LMS has been the subject of numerous studies in recent decades [1, 3, 8, 9], considerable challenges remain open [1]. The study developed by [4] sought to establish the relationship between ergonomics and usability in the context of e-learning. An Systematic Mapping (SM) was developed by [5] to identify publications related to the usability of e-learning systems in the mobile context, specifically on mobile devices. In another SM, [10] suggests an update to the work developed by [5]. The authors observed an evolution in the approaches and evaluation techniques of m-learning LMS. However, they noticed the absence of a comprehensive framework or methodological approach to address UX, usability, and the pedagogical aspects of mobile educational software. In response to this, they proposed a framework to evaluate mobile LMS. Subsequently, [3] developed an SM to identify publications that evaluate both desktop and mobile LMS applications from a usability and UX perspective.

Methodological approaches used to evaluate LMS often rely on interface evaluation [11] and approaches focusing on the pedagogical domain [12–14]. Using more sophisticated evaluation strategies, such as AI, to investigate the usability and UX is still uncommon [1, 3, 8, 9]. Heuristic evaluation based on predefined criteria is by far the most common method. These criteria are supported by guidelines such as the ISO usability [15, 16] and W3C [17] standards and Nielsen's "10 Usability Heuristics for User Interface Design" [11, 18]. Research suggests that many usability evaluations conducted on LMS over the past decade are based on those heuristics (10 HU-JN) [6, 19].

One of the main challenges for adopting AI-based approaches for heuristic evaluation of interfaces is defining the set of indicators that can be converted into datasets for training AI models. Considering this challenge, we have designed a methodological approach to investigate a predefined set of 800 Ui based on Nielsen's heuristics (10 HU-JN) to be used in the development of datasets for machine learning (ML) models. This approach supported the subsequent phase, where several AI models were tested to contribute to the process of evaluating the usability of LMS interfaces. The remainder of the article is organized as follows: Sect. 2 presents the theory used to sustain this study; Sect. 3 details the methodology adopted in our study; Sect. 4 discusses the results obtained; and finally, Sect. 5 presents the conclusion.

2 Background and Related Work

Heuristic evaluations (HE) reveal usability issues early in the design process. Combined with other testing methods, they can be more cost-effective than testing involving real users [20–22]. One challenging aspect of HE is its intrinsic relationship with the evaluator's experience and knowledge [22]. The inspection process relies on assumptions made by the expert regarding what constitutes "good" or "poor" usability. Despite the extensive research conducted over many years to establish the ten heuristics (10 HU-JN), there are inherent subjective elements to consider in the evaluation process [20, 23]. HE involves adopting a set of usability criteria or indicators that identify potential interface issues, often rooted in the system's underlying structures, development, and design [24]. The challenge of implementing AI-based approaches lies in finding a set of Uis that can reduce the impact of subjectivity inherent to the HE process.

Ui groups have been used since the 1960s. Initially, Uis were associated with studies about a system's biomechanical and psychophysiological responses [25]. In the 1970s, concerns regarding usability criteria emerged, leading to the creation of ISO standard 9241–11 [26]. Works by Jordan [27] and Nielsen on Usability Inspection Methods [11] and Usability Engineering [28], as well as Dominique and Scapin's research on ergonomic quality criteria for interactive systems [29], were landmarks of this period. In addition to the ISO standard, other standardizations for web content development would later emerge, such as the W3C/WAI Web Content Accessibility Guidelines [17]. The use of criteria to evaluate system interfaces has been the subject of research in seminal works over the past decades [11, 27, 29–39]. Since then, an increasing number of evaluation criteria have been proposed, resulting in a considerable set of Uis that, in some cases, may overlap [25, 40–44]. In the field of LMS studies, research point to the use of diverse sets of Uis, techniques, and different methodological approaches adopted to

evaluate the usability of these systems [1, 3–5, 5, 6]. Despite the proposals of different frameworks in recent years for assessing LMS usability, researchers and professionals still lack consensus regarding a unified evaluation model, and initiatives that employ AI-based approaches are still relatively scarce [3, 6].

The definition of an AI-based approach for HE implies considering problem definition and characterization, AI algorithms and models, data types, input variables, datasets, training and testing data, features, labels, among others [45]. AI encompasses two major general approaches: knowledge-based and statistical data-based. The former is referred to as GOFAI—"*Good Old-Fashioned Artificial Intelligence*" aiming to encode expert knowledge into software. Although in this case, the machine is not capable of learning on its own [46]. In statistical-based Machine Learning, methods are developed to enable the machine to learn through mathematical techniques. [46]. To adopt an AI-based solution, it is necessary to define the problem to be solved, which will be converted into AI tasks [47]. This process involves identifying algorithms to construct mathematical-statistical models that can perform those tasks [48]. An algorithm is a well-defined set of instructions designed to accomplish a specific goal using input data. It operates by executing a series of finite steps. In traditional computer programming, a known and predefined algorithm performs a particular task (Input Data + Model → Output) [48]. In the context of AI, the algorithm serves as a recipe or set of instructions for applying an AI technique to obtain an artificial intelligence model [45]. The algorithm utilizes available data to construct a data-driven model. It's important to note that different data inputs can lead to different models, even when using the same algorithm [48]. In the realm of AI, the specific model is not predetermined but is learned and generated by the algorithm based on the provided data [45]. This process allows the algorithm to adapt and derive insights from the data, ultimately producing a model tailored to the specific problem or task at hand [45]. Data can be categorized as either discrete or continuous. Discrete data corresponds to categorical values. Examples of discrete data include binary choices like "yes" or "no" and categorical options like "A," "B," or "C." On the other hand, continuous data represents values within a range or continuum. For instance, it can involve measuring the number of events occurring within a specific unit of time [45].

In programming, a variable is a storage location that holds a value and can be modified during the execution of an algorithm [49]. Two types of dependency relationships between variables are crucial: independent and dependent variables [48]. The independent variable exists independently and serves as the input data used to train the AI model. It is typically manipulated or controlled by the algorithm to observe its impact on the dependent variable [45]. The dependent variable, on the other hand, is the one that needs to be predicted or estimated, and its value is dependent on another variable. The algorithm aims to learn the relationship between the independent and dependent variables through the training process, allowing it to make predictions or draw conclusions based on new input data [45, 47].

A dataset is a vital resource utilized for training, testing, and validating AI models. Within the field of machine learning, a dataset comprises input features alongside their corresponding output labels or target values [47]. These datasets can be sourced from a wide range of origins, including experimental studies, real-world observations, or artificially generated simulations. The quality and efficacy of an AI model are often

strengthened by the presence of datasets that not only exhibit diversity but also serve as representative samples of the underlying domain [45]. The inclusion of comprehensive and representative data empowers the model to faithfully capture the intricacies and complexities of real-world scenarios, leading to enhanced performance and broader applicability [50].

In general, datasets are partitioned into training and test data, enabling the evaluation and comparison of different ML models [50]. The data is typically split into an 80/20 ratio, with 80% used for training the model and 20% reserved for testing purposes [50]. Each individual value within a dataset is commonly referred to as a data point, analogous to a row in a table that presents various patient data such as blood pressure, red and white blood cell counts, and other relevant attributes [46, 48].

Within a dataset, features correspond to the input variables that describe each available example or data point, while labels pertain to the output variable or target value. When a training dataset is employed to refine a model, it undergoes a training process often referred to as an epoch [45]. During this iterative process, the model learns from the training dataset, adjusting its internal parameters to optimize its ability to make accurate predictions or classifications based on the input features and their associated labels [46, 48].

Lastly, the concept of informational entropy holds significance in the realm of AI models as it quantifies the informational organization within a given system or dataset. Introduced by Shannon, entropy represents the average amount of information contained within a message or signal, typically measured in bits. It is calculated based on the probability distribution of each potential outcome within the system or dataset [51].

Machine Learning is fundamentally rooted in the utilization of existing data, making it data-driven at its core [52]. Unlike traditional rule-based systems, ML techniques do not rely on predefined rules but instead learn from examples derived from available data and make predictions or decisions based on incoming data. ML encompasses three primary types: supervised learning (SL), unsupervised learning (UL), and reinforcement learning (RL) [46, 48]. Supervised learning is further classified into two main categories: classification and regression. Classification involves using labeled categorical data to produce categorical results, while regression involves utilizing unlabeled data with continuous values. On the other hand, unsupervised learning deals with datasets that consist of numerous unlabeled data points [48]. In the context of this research, supervised learning was selected as the starting point due to its ability to create a dataset with appropriate labels. This choice is particularly significant in a relatively new research domain where large, labeled datasets specifically related to LMS are not yet readily available.

3 Methodology

An investigation was conducted to explore a set of Uis for the heuristic evaluation of LMS using ML models. This phase was based on a collection of 800 Uis obtained through a literature review of the theoretical baseline developed by Jakob Nielsen [19]. The collection of Uis can be found at: https://jc7.co/p2dg231. The investigation of Uis for AI was carried out using a novel systematic process that aimed to identify the parameters

necessary to utilize Uis as a foundation for constructing datasets in AI models. The systematization process involved the following stages, which will be discussed in the following sessions.

Working Document: A spreadsheet based on the set of 800 Uis previously defined, available at: https://jc7.co/p2dg232.

Subjectivity Factor (SF) Identification: Establishment of an arbitrary classification based on the proximity of the usability indicators to aspects that are difficult to quantify or measure, as determined by the interface analyst's evaluation. The SF aimed to determine if there were chances of different evaluators interpreting a criterion differently. The primary assumption adopted was to consider SF equal to 1 (positive subjectivity) whenever an adjective was encountered.

SF Filter: A filter was employed to acquire Ui whose SF equaled zero. The objective was to identify indicators that could be studied through simplified, automated evaluation without the need to adopt aspects inherent to the concepts of "good" or "bad" (in this case, with negative subjectivity – equal to 0). Graphics and data are available here: https://jc7.co/p2dg233 on page 01.

Definition of Data Types Associated with Uis: A suggestion for data types related to the analysis context of Uis by AI based on the following classification – Interval Data – available in the form of any intervals; Binary Data – related to the simple binary identification of the type Yes/No; On/Off; Available/Unavailable, etc.; and N/A – when it was not possible to define a data classification through immediate analysis of the context in which the Ui would be inspected. Graphics and data are available here: https://jc7.co/p2dg233 on pages 11 and 12.

Examples Related to the Identified Data: Include situations such as "conditional display occurs/conditional display does not occur/"; "status indicated/status not indicated"; "feedback occurs/feedback does not occur"; "notification occurs/notification does not occur," among others. A restrictive classification was not defined for the examples since the purpose was to identify how the system "communicated" through the data to inform that a certain Ui had been satisfied. Graphics and data are available here: https://jc7.co/p2dg233 on pages 13 and 14.

Types of Data Related to AI Models: Based on discrete data – can assume only several countable values; and continuous data – can assume all possible values within a certain range. The objective was to find indications of potential relationships between the types of data and AI tasks suggested in the following phases. Graphics and data are available here: https://jc7.co/p2dg233 on pages 15 and 16.

Proposal for an Algorithmic Evaluation of Ui by AI: We adopted a work protocol inspired by the concept of "Designer's Proxy" from Semiotic Engineering (SE), according to which the interface of a system acts as a communicational artifact on behalf of the designer in the user interaction process [53]. In the context of this study, we considered that "AI would help the evaluator inspect the system interface"; therefore, it would act as a kind of "Evaluator's Proxy". The meta-communicational approach of SE advocates

that certain assumptions must be met for everything to go well. Following this principle of thought, we considered it necessary to verify what assumptions would be required for AI to contribute to making the interface evaluation process easier for the evaluator. The approach was named the EU-Machine Protocol, and it consisted of the simple exercise of asking oneself aloud: "*What do I – as a machine, alone – need to do to evaluate this usability indicator without asking for help from a human being?*". During the procedure, numerous actions required to inspect the indicator on the interface were enunciated formally. These vernacular expressions were then transcribed into algorithmic form. The "Portugol" language (known as structured Portuguese in Pascal programming) was utilized as a logical reference to discern the necessary steps for evaluating the Ui on the interface. This information was subsequently converted into AI-driven analysis statements. Some indicators proved unfeasible to analyze or excessively intricate, primarily due to factors about their limited level of detail in the description, subjectivity, or complexity. These indicators were flagged and served as a foundation for identifying areas where indicator decomposition and/or system adaptation were deemed necessary. In total, 256 indicators could be examined from the AI perspective (32% of the originally proposed 800 indicators) based on our analysis.

For instance, the algorithm applied to Ui-40 (1 – System Status Visibility (Heuristic) COMMUNICATION (Level 1) ALERTS (Level 2) – executed action (Level 3)) corresponded to the following statement: "*Define the action to be monitored (step 1). Define an identifier for the action's state (step 2). Define a flag to capture the action's state (step 3). Define an element to display alerts based on the flag's state (step 4). Define a flag for the element's state displaying the alert (step 5). Compare the action's flag state with the alert's flag state (step 6). Validate if the states relate as expected.*" The same procedure was applied to all usability indicators. Proposed algorithms can be found here: https://jc7.co/p2dg232.

Number of Process Steps (NPS): A variable indicating the count of steps necessary to implement the suggested algorithm minus one. Analyzing the statement from the previous example, we have NEPS = 6 (7 steps – 1). Graphics and data are available here: https://jc7.co/p2dg233 on pages 17 and 18.

Decomposition or Disambiguation Requirement (DDR): The aim was to investigate whether the Ui was defined clearly, descriptively, and objectively without needing further technical specifications, elaboration, or disambiguation of the analysis proposal expressed in its statement. The variable DDR could take on two possible values. DDR = 0 indicates that the Ui did not require decomposition or disambiguation, and DDR = 1 indicates that the Ui required decomposition or disambiguation. For example, Ui-15 was defined with DDR = 1. The statement "*conditional display of GUI elements*" was deemed insufficiently descriptive and objective, as the term "conditional" does not explicitly define under which conditions the display should occur, and the term "GUI elements" does not objectively indicate which interface elements should be conditionally displayed.

On the other hand, Ui-122 was marked with DDR = 0. The statement "*LIMITS/ALERTS/character quantity*" was considered sufficiently descriptive and objective, as the terms "ALERTS" and "LIMITS," combined with the term "character quantity," indicate the need to confirm the presence of that Ui, in the form of an alert,

that demonstrates restricted character insertion in a specific field of a form. Graphics and data are available here: https://jc7.co/p2dg233 on pages 6 and 7.

System Adaptation and/or Update (SAU): Based on the Ui statement and the suggested algorithm, the need for system adaptation to be inspected from an AI approach was determined. AI tasks involve creating a dataset to train and test models. In some cases, data collection must be done manually, either through visual inspection or, ideally, through automated techniques such as reading logs, code inspection, available browser cookies, and checking CSS and HTML properties, among other tasks. However, there are cases where automated data collection is not possible, requiring initial system adaptation to provide specific functionalities. For example, recording data in files, and storing information in databases, among others. The variable could take on two possible values. SAU = 0 – to express that potential AI inspection of the Ui did not require system adaptation and/or update, and SAU = 1 – to express the need for system adaptation and/or update. For example, Ui-424 was considered SAU = 1. Its AI analysis statement (Algorithm) is "*validate the existence of a placeholder in the form field through source code analysis*". The justification relies on the idea that based on a trivial source code inspection, it is possible to confirm the existence of a "placeholder" tag in the HTML library (https://www.w3schools.com/tags/att_input_placeholder.asp). Therefore, there is no need to adapt the system for the information to be available, although methods can be considered to automate data collection to reduce or eliminate manual work.

Ui-352 has an SAU = 0 and has the following AI analysis statement (Algorithm): "*Define the current state of the system based on storage via a database, files, cookies, or another strategy. Define the current page. Validate if changes in data have already been stored. If they are not stored, save them in a temporary virtual environment. Define a virtual space to store the data. Validate the filling of the virtual space with the stored data. Associate a flag to identify if the action was satisfactory*". The justification is that the algorithm statement sought to address the usability indicator's specified requirement, considering strategies that required altering the system's structure by adding functionalities such as storing data in databases, and files, among others, and defining and associating a flag to identify the monitored action. Graphics and data are available here: https://jc7.co/p2dg233 on pages 23 and 24.

Complexity Factor (CF): An arbitrary scale indicating the degree of effort required to apply the proposed algorithm. The scale considered the number of steps required for AI evaluation, the need for Ui disambiguation, and the system adaptation requirement. The established mathematical relationship adopted weights and examples can be found in supplementary material at the end of this section. Graphics and data are available here: https://jc7.co/p2dg233 on pages 9 and 10.

Machine Learning Type (MLT): The aim was to identify the type of ML, whether it falls under the categories of supervised (SL), unsupervised (UL), or reinforcement learning (RL), based on the Ui statement and the proposed approach for inspecting it from the AI perspective, as outlined in the designed algorithm. The majority of MLT was classified as SL.

AI Task: The objective was to identify tasks such as classification and linear regression, typically adopted as fundamental approaches within ML. Graphics and data are available here: https://jc7.co/p2dg233 on pages 19 and 20.

Decision Purpose: The purpose was to identify the decision objective of the suggested AI task for inspecting the Ui, based on the prediction and diagnosis categories. This classification followed the analysis suggestion proposed by the "AI Map" document developed by the Japanese Society for Artificial Intelligence [54]. Graphics and data are available here: https://jc7.co/p2dg233 on pages 21 and 22.

Data Processing: The data and graphs were processed in the Power BI platform and are presented below.

4 Discussion

This section presents the results obtained for the Ui classification stage, later used in another study to develop a methodology to adopt AI as part of the interface inspection process based on heuristic evaluation. Only Uis whose FS was determined to be equal to zero were analyzed. The following sections discuss the results found.

Types of Associated Data: Approximately 82.95% (292) of the evaluated Uis could be analyzed through the collection of binary data (YES/NO) distributed across all heuristics. 17% (60) of the Uis were associated with interval data types, which were more prevalent in heuristics *5 – Error Prevention, 4 – Consistency and Standards, and 8 – Static and Minimalistic Design.* Binary data can be associated to diagnose the interface, where scrutiny is performed to validate the existence of a specific element of interest through simple inspection of its presence in the expected location.

Examples Related to the Data: The most recurring example across the heuristics was "Indicated Status/Not Indicated Status" (274 occurrences – 78%), except for heuristics *5 – Error Prevention and 4 – Consistency and Standards*, where the most recurring example was "Records variation in item quantity" (38 occurrences – 10%). It is possible to establish a relationship between the observed pattern in these two heuristics and the associated data type, which was more strongly linked to interval data for both. Other relevant examples include "Records variation in item quantity" and "Warning occurs/Warning does not occur". The complete list of examples is available at: https://jc7.co/p2dg233 on page 13.

Types of Input Data: Approximately 85.22% (300) of the evaluated Uis were characterized as being related to discrete values, while approximately 14.77% (52) of the Uis were associated with continuous values. Discrete data may be associated with classification tasks in AI. These values are predominant across all heuristics, except for heuristics *5 – Error Prevention and 4 – Consistency and Standards*, which have Uis where the input data for the AI model is of continuous type in more than 50% of the cases, possibly because their indicators were related to interval data in several situations.

Evaluation Algorithm: It was utilized to elucidate the requirements for using AI models to assist in evaluating Uis in interfaces. The algorithm contributed to identifying the

necessary inputs to perform the procedure and possible expected outputs at the end of the inspection. The statements from this phase were essential in highlighting the variables NPS, SAU, and CF.

Number of Process Steps: The most frequent number of steps corresponds to 2, 3, 4, 6, and 5. The number of steps is directly related to the proposed algorithm. The maximum number found was 13, and the minimum was two within a range established between 1 and 15. It is expected that new rounds of algorithmic analysis will make the process more precise and detailed, increasing the number of steps and expanding the technical requirements for implementing the proposals. The task of proposing algorithms to evaluate Uis requires the involvement of professionals from design, web development, and artificial intelligence. The presented results reflect only one round of algorithm analysis to investigate the indicators, and further rounds would likely impact the number of steps, increasing them to further clarify the proposed use of AI approaches. Another point is the practical implementation of the algorithms. The execution of the proposed tasks in an experimental phase, based on initial prototypes, may demonstrate the need to modify the algorithm and, consequently, the number of steps, as well as establish the need to relate and evaluate indicators together in an inseparable manner, to see if different results are obtained.

Decomposition/Disambiguation of usability indicators: The objective was to identify if the available information was sufficient to directly inspect the interface or if more details would be necessary. Approximately 86% (277) of the evaluated Uis required some form of a declarative specification to make them less ambiguous, more descriptive, and direct. Approximately 13% (45) of the Uis had statements considered direct and objective. The elimination criterion adopted was an FS equal to 0. Among the 352 Uis with an FS determined to be equal to zero, 96 were subsequently classified as ambiguous or requiring significant decomposition, which could potentially impact the suggestion of other AI-related aspects. During the study, it was observed that to evaluate interfaces with AI-based approaches, adaptations need to be made to the platforms to capture data that will be used to build datasets for the AI models. Therefore, it is necessary to prepare the systems to mark points of interest and track user actions considering privacy and security issues, among other modifications. For example, in several indicators, the use of flags (markers) in the source code was suggested in the algorithm proposals. However, most systems are not yet geared toward data collection in this manner. Some studies identified in the exploratory phase of this research demonstrate that the evaluation of LMS interfaces, for example, was done through log analysis, database analysis, or even directly by manually collecting data at specific "moments" when researchers examined the system [1]. Therefore, automated data collection to feed evaluation systems should be an objective to be considered by organizations that wish to evaluate their platforms in an automated manner.

Complexity Factor: The implementation of AI-based approaches to inspect interfaces based on the Uis from the obtained corpus was considered high for approximately 44% (157) of the indicators, medium for approximately 49% (173) of the indicators, and low for 6.25% (22) of the studied Uis. Overall, the implementation of AI approaches to evaluate Uis in a heuristic evaluation of interfaces is considered high or medium for

over 70% of the studied Uis for each of the heuristics. This is mainly due to the need for system adaptation for data collection, in addition to the inherent complexity of algorithm implementation.

AI Task: Approximately 84% (296) of the studied Uis were associated with classification tasks, while 16% (56) were associated with regression tasks. The relevance of the classification task may be related to the types of statements found in the Uis, where the analysis aims to confirm the presence of certain objects of interest in the interface. In other words, it is about determining whether an analyzed element is or is not where it should be, expressed as it should be, or possesses the expected characteristics, among other situations that allow for relatively easy grouping to classify the indicators. For example, in the case of Ui-40 (1 – System Status Visibility (Heuristic) – COMMUNICATION (Level 1) – ALERTS (Level 2) – action executed (Level 3)), it aims to verify if a specific alert was displayed in the interface when a certain action was performed. In this context, either the alert was displayed, and the usability indicator evaluation allows for satisfactory classification, or vice versa; if it does not occur, it is impossible to confirm the alert's presence, indicating an unsatisfactory situation. Thus, it is possible to establish two sets and classify the indicators based on the result using a classification process. While identifying AI tasks to investigate the Uis, it was observed that, in most cases, it is necessary to combine indicators for a heuristic evaluation of the interface with AI support to become more meaningful. This is because there are complex relationships between the Ui established by the evaluator during the inspection, making the inspection richer but also challenging and complex. This presents an exciting opportunity for adopting AI-supported practices that can automate, simplify, and reduce the cognitive effort of the evaluator in establishing relationships between different analyzed features simultaneously and demonstrates how the pathway to have a complete AI technique along the process is quite challenging now. Opportunities for applying simple and multiple linear regression models and classification tests were identified in the subsequent phase of the research. Linear regression is an interesting task to establish relationships between different features in a dataset, as it provides a simple, easily interpretable, flexible, and fast learning method that involves a simple and linear relationship between input and output variables. Methods like linear regression offer a good opportunity for initial and exploratory studies where there is an interest in understanding the essential relationships between variables. Multiple linear regression, on the other hand, can identify more elaborate relationships precisely because it incorporates multiple variables as features in the learning process.

Purpose of the Technique: Most Uis were associated with the diagnosis. This can be explained by the propensity of heuristic evaluation to diagnose the interface, validating attention to certain requirements established in the Ui statements.

The adoption of AI approaches to support the process of interface inspection should consider complex actions and events occurring in the interaction environment, some of which are difficult to track, capture, and manage through automated data collection processes. In several analyzed situations, there are features related to the processing and storage of data in databases or browser caches to record the various events that occur in the system based on user actions or platform events. These actions occur on the system's front end but require logs to be stored locally on the user's computer via

cookies or other formats and then transferred to a database for analysis. At the same time, measures need to be taken to ensure that the infrastructure of the environment can have checkpoints for the indicators, using flags or markers that can identify the data of interest. In various situations, these and other contexts require changes and adaptations in system development so that AI can be adopted to contribute to inspecting and evaluating interfaces.

During the classification process, it was observed that AI could be a strategy to address specific interface analysis problems. However, several Uis were still open-ended and lacked clear explanations about what should be assessed in the interface. Our perception regarding the use of AI techniques for Ui evaluation, aiming to inspect interfaces, is that the effort to implement such approaches is significantly greater than making the necessary adaptations in systems to ensure that interfaces meet the minimum requirements for automated analysis, especially in terms of providing the necessary data for building AI models. Therefore, we advocate for the construction and/or adaptation of systems oriented toward AI-based evaluation from the ground up.

Initial analyses of this exploration demonstrated that a significant portion of Uis can be considered from a binary perspective. This is plausible, considering that they would already simplify the usability concepts that are expected to be evaluated. It is worthwhile to reconsider Uis or even new heuristics that look into the internal structures of systems, considering their architecture, source code, and backend, in areas that are not typically inspected but have an impact on system usability. When considering an interface evaluation process in the traditional approach, we are "looking" at the interface and identifying issues that resulted from a development process that supposedly did not meet the expected requirements. However, by considering this internal perspective at a deeper level, it may be possible to address problems before they impact the surface (interface).

In some cases, adopting AI approaches will not require modifying systems to obtain the necessary data for creating models, but it requires further studies considering this approach. During the collection and classification of Ui's, in several situations, it was observed, for example, that it was necessary to validate the presence of a marker (a type of flag) in the source code without requiring adaptations/updates to the software structure, only requiring finding references that indicate the presence of a specific element expressed in the source code. Experimental tests supported by the adoption of a methodology oriented towards the evaluation of Uis with the help of AI were conducted in the following stages of the research. However, it is worth noting that this analysis, in many cases, is still performed either fully or partially manually. Therefore, providing external or internal resources to the analyzed platforms to enable data collection can be a favorable factor. We also found that to adopt an AI approach to reduce the evaluator's effort in repetitive activities it is necessary to design more restrictive statements to reduce the chances of different interpretations; otherwise, evaluators will constantly need to be involved in interpreting the statements. In the future, it will be necessary to adapt systems to emit certain signals about the user experience, allowing data to be collected while respecting privacy rules.

5 Conclusion

During the classification process, it was observed that AI can be a strategy to be adopted to address specific problems in interface analysis. However, several Uis were still open-ended and lacked clear explanations about what should be assessed in the interface. Our perception regarding the use of AI techniques for Ui evaluation, to inspect interfaces is that the effort to implement such approaches is significantly greater than making the necessary adaptations in systems to ensure that interfaces meet the minimum requirements for automated analysis, especially in terms of providing the necessary data for building AI models. Therefore, we advocate for the construction and/or adaptation of systems that are oriented towards AI-based evaluation from the ground up. Initial analyses of this exploration demonstrated that a significant portion of Uis can be considered from a binary perspective. This is plausible, considering that they would already simplify the usability concepts that are expected to be evaluated. It is worthwhile to reconsider Uis or even new heuristics that look into the internal structures of systems, considering their architecture, source code, and backend, in areas that are not typically inspected but have an impact on system usability. When considering an interface evaluation process in the traditional approach, we are "looking" at the interface and identifying issues that resulted from a development process that supposedly did not meet the expected requirements. However, by considering this internal perspective at a deeper level, it may be possible to address problems before they impact the surface (interface). In some cases, adopting AI approaches does not require modifying systems to obtain the necessary data for creating models. In several situations, it was observed, for example, that it was necessary to validate the presence of a marker (a type of flag) in the source code without requiring adaptations/updates to the software structure, only requiring finding references that indicate the presence of a specific element expressed in the source code. However, it is worth noting that this analysis, in many cases, is still performed either fully or partially manually. Therefore, providing external or internal resources to the analyzed platforms to enable data collection can be a favorable factor. We also found that to adopt an AI approach to reduce the evaluator's effort in repetitive activities, it is necessary to design more restrictive statements to reduce the chances of different interpretations; otherwise, there will be a constant need to involve evaluators to interpret the statements. In the future, it will be necessary to adapt systems to emit certain signals about the user experience, allowing data to be collected while respecting privacy rules.

Acknowledgments. This study was supported by UNIDCOM under a Grant by the Fundação para a Ciência e Tecnologia (FCT) No. UIDB/DES/00711/2020 attributed to UNIDCOM/IADE – Unidade de Investigação em Design e Comunicação, Lisbon, Portugal.

References

1. Júnior DGS (2019) Usability and User Experience Evaluation of Learning Management Systems. In: DCC. Unidcom, IADE, Lisbon
2. Zaharias P, Koutsabasis P (2011) Heuristic evaluation of e-learning courses: a comparative analysis of two e-learning heuristic sets. Campus-Wide Inform. Syst. 29(1):45–60

3. Takashi Nakamura W, Harada Teixeira de Oliveira E, Conte T (2017) Usability and user experience evaluation of learning management systems – a systematic mapping study. In: Proceedings of the 19th international conference on enterprise information systems [Internet]. SCITEPRESS – Science and Technology Publications, Porto, Portugal; [cited 2020 Jun 20], pp 97–108. https://doi.org/10.5220/0006363100970108
4. Freire LL, Arezes PM, Campos JC (2012) A literature review about usability evaluation methods for e-learning platforms. Work 41:1038–1044
5. Cota CXN, Díaz AIM, Duque MÁR (2014) Evaluation framework for m-learning systems: current situation and proposal. In: Proceedings of the XV international conference on human computer interaction – Interacción '14 [Internet]. Puerto de la Cruz, Tenerife, ACM Press, Spain [cited 20 Jun 2020], pp 1–3. http://dl.acm.org/citation.cfm?doid=2662253.2662265
6. Júnior DGS, Hernández-Ramírez R, Estima J (2022) Systematic mapping of methods used to evaluate the usability and UX of learning management systems. In: Martins N, Brandão D (eds) Advances in design and digital communication II: Proceedings of the 5th international conference on design and digital communication, Digicom 2021, 4–6 Nov 2021, Barcelos, Portugal. Springer International Publishing, Cham, pp 122–133. https://doi.org/10.1007/978-3-030-89735-2_11
7. Hassenzahl M, Law E, Ebba H (2006) User Experience – Towards a unified view. UX WS NordiCHI'06: COST294-MAUSE 161
8. Cantabella M, López B, Caballero A, Muñoz A (2018) Analysis and evaluation of lecturers' activity in learning management systems: subjective and objective perceptions. Interact Learn Environ 26(7):911–923
9. Mtebe JS, Kissaka MM (2015) Heuristics for evaluating usability of Learning Management Systems in Africa. In: 2015 IST-Africa conference [Internet]. Lilongwe, IEEE, Malawi [cited 20 Jun 2020], pp 1–13. http://ieeexplore.ieee.org/document/7190521/
10. Navarro CX, Molina AI, Redondo MA, Juarez-Ramirez R (2016) Framework to evaluate m-learning systems: a technological and pedagogical approach. IEEE R Iberoamericana Tecnologias Aprendizaje. 11(1):33–40
11. Nielsen J, Mack RL (eds.) Usability inspection methods. Wiley, New York (1994), 413 p.
12. Hosie P, Schibeci R, Backhaus A (2005) A framework and checklists for evaluating online learning in higher education. Assess Eval High Educ 30(5):539–553
13. Mtebe JS, Kissaka MM (2015) Heuristics for evdaluating usability of Learning Management Systems in Africa. In: 2015 IST-Africa Conference [Internet]. IEEE, Lilongwe, Malawi [cited 2019 Jun 4], pp 1–13. http://ieeexplore.ieee.org/document/7190521/
14. Oztekin A, Kong ZJ, Uysal O (2010) UseLearn: a novel checklist and usability evaluation method for eLearning systems by criticality metric analysis. Int J Ind Ergon 40(4):455–469
15. Part 11: Usability: Definitions and concepts (ISO 9241-11:2018). BSI Standards Publication (2018)
16. Standard I. ISO 9241-210 (2010)
17. W3C Web Accessibility Initiative (WAI) (2018) Web Content Accessibility Guidelines (WCAG) Overview [Internet]. Web Accessibility Initiative (WAI). [cited 7 Jan 2019]. https://www.w3.org/WAI/standards-guidelines/wcag/
18. Nielsen J (1994) Enhancing the Explanatory Power of Usability Heuristics
19. Junior DGS, Hernández-Ramírez R, Estima J (2023) Systematic mapping of criteria used to evaluate the usability and UX of learning management systems. In: Perspectives on design and digital communication: research, innovations and best practices. Springer, Portugal
20. Barnum CM (2021) Usability testing essentials: ready, set...test! 2nd edn. Morgan Kaufmann, Amsterdam 448 p
21. Nielsen J (1994) Heuristic Evaluation: How-To: Article by Jakob Nielsen [Internet]. Nielsen Norman Group. [cited 27 Sep 2022]. https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/

22. Wong E (2022) Heuristic Evaluation: How to Conduct a Heuristic Evaluation [Internet]. The Interaction Design Foundation. [cited 25 Sep 2022]. https://www.interaction-design.org/literature/article/heuristic-evaluation-how-to-conduct-a-heuristic-evaluation
23. Heuristic evaluations vs. usability testing [Internet] (2001) Human Factors International. [cited 27 Sep 2022]. http://www.humanfactors.com
24. Carroll JM (1997) Human-computer interaction: psychology as a science of design. Annu. Rev. Psychol. 48:61–83
25. Brangier E, Desmarais M, Alexandra N, Tep SP (2015) Évolution de l'inspection heuristique: vers une intégration des critères d'accessibilité, de praticité, d'émotion et de persuasion dans l'évaluation ergonomique. Journal d'Interaction Personne-Système 4(1):16
26. Brangier E, Desmarais MC (2014) Heuristic inspection to assess persuasiveness: a case study of a mathematics e-learning program. In: Marcus A (ed) DUXU 2014, vol 8517. LNCS. Springer, Cham, pp 425–436. https://doi.org/10.1007/978-3-319-07668-3_41
27. Jordan PW (1998) An introduction to usability. Taylor & Francis, London, Bristol, Pa 120 p
28. Nielsen J (2010) Usability engineering. Nachdr. Kaufmann, Amsterdam 362 p
29. Scapin DL, Bastien JMC (1997) Ergonomic criteria for evaluating the ergonomic quality of interactive systems. Behav Inform Technol 16(4–5):220–231
30. Bennett JL (ed.) (1984) Visual display terminals: usability issues and health concerns. Prentice-Hall, Englewood Cliffs, N.J. 297 p
31. Scapin DL (1990) Organizing human factors knowledge for the evaluation and design of interfaces. Int J Human-Comput Interact. 2(3):203–229
32. Shneiderman B, Plaisant C (2004) Designing the user interface: strategies for effective human-computer interaction, 4th edn. Pearson/Addison Wesley, Boston, p 652
33. Shackel B (2009) Usability – Context, framework, definition, design and evaluation. Interact Comput 21(5–6):339–346
34. (2016) Advances in ergonomics in design. Springer Berlin Heidelberg, New York, NY
35. Marcus A, Wang W (eds) (2017) Design, user experience, and usability: designing pleasurable experiences: 6th international conference, DUXU 2017, Held as Part of HCI International 2017, Vancouver, BC, Canada, July 9-14, 2017, Proceedings, Part II. Springer International Publishing, Cham
36. Marcus A, Wang W (eds) (2017) DUXU 2017, vol 10288. LNCS. Springer, Cham. https://doi.org/10.1007/978-3-319-58634-2
37. Marcus A, Wang W (eds) (2017) DUXU 2017, vol 10290. LNCS. Springer, Cham. https://doi.org/10.1007/978-3-319-58640-3
38. (2017) Advances in usability and user experience. Springer Berlin Heidelberg, New York, NY
39. Ahram TZ, Falcão C (eds) (2019) Advances in usability, user experience and assistive technology. In Proceedings of the AHFE 2018 international conferences on usability & user experience and human factors and assistive technology, held on 21–25 July 2018, in Loews Sapphire Falls Resort at Universal Studios, Orlando, Florida, USA [Internet]. Springer International Publishing, Cham [cited 7 Jan 2019]. (Advances in Intelligent Systems and Computing; vol 794). https://doi.org/10.1007/978-3-319-94947-5
40. Michel G, Brangier E, Brun M (2014) Ergonomic principles to improve the use of cognitive stimulation systems for the elderly: a comparative study of two software tools. In: Stephanidis C, Antona M (eds) Universal access in human-computer interaction. aging and assistive environments: 8th international conference, UAHCI 2014, held as part of HCI international 2014, Heraklion, Crete, Greece, 22–27 June 2014, Proceedings, Part III. Springer International Publishing, Cham, pp 147–154. https://doi.org/10.1007/978-3-319-07446-7_15
41. Némery A, Brangier E (2014) Set of guidelines for persuasive interfaces: organization and validation of the criteria. J Usability Stud 9(3):24

42. Duczman M, Brangier E, Thévenin A (2016) Criteria based approach to assess the user experience of driving information proactive system: integration of guidelines, heuristic mapping and case study. In: Rebelo F, Soares M (eds) Advances in ergonomics in design: proceedings of the AHFE 2016 international conference on ergonomics in design, 27–31 July 2016, Walt Disney World®, Florida, USA. Springer International Publishing, Cham, pp 79–90. https://doi.org/10.1007/978-3-319-41983-1_8
43. Urrutia JIG, Brangier E, Cessat L (2017) Is a holistic criteria-based approach possible in user experience? In: Marcus A, Wang W (eds) DUXU 2017, vol 10288. LNCS. Springer, Cham, pp 395–409. https://doi.org/10.1007/978-3-319-58634-2_29
44. Urrutia JIG, Brangier E, Senderowicz V, Cessat L (2018) Beyond "usability and user experience", towards an integrative heuristic inspection: from accessibility to persuasiveness in the UX evaluation: a case study on an insurance prospecting tablet application. In: Ahram T, Falcão C (eds) Advances in usability and user experience: proceedings of the AHFE 2017 international conference on usability and user experience, 17–21 July 2017, The Westin Bonaventure Hotel, Los Angeles, California, USA. Springer International Publishing, Cham, pp 460–470. https://doi.org/10.1007/978-3-319-60492-3_44
45. Hurbans R (2020) Grokking artificial intelligence algorithms. Manning Publications, New York, p 362
46. Frankish K, Ramsey WM (eds) (2014) The Cambridge handbook of artificial intelligence. Cambridge University Press. https://doi.org/10.1017/CBO9781139046855
47. Domingos P (2015) The master algorithm: how the quest for the ultimate learning machine will remake our world. Basic Books, a member of the Perseus Books Group, New York 329 p.
48. Russell SJ, Norvig P, Davis (2010) Artificial intelligence: a modern approach, 3rd edn. Prentice Hall, Upper Saddle River (Prentice Hall series in artificial intelligence), 1132 p
49. Liang YD (2015) Introduction to Java programming, 10th edn. Pearson, Boston, p 1320
50. Géron A (2022) Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, Inc., 1351 p
51. Shannon CE (1948) A mathematical theory of communication. The Bell Syst Tech J 27:379–423, 623–656
52. Mitchell TM (1997) Machine learning. McGraw-Hill, New York (McGraw-Hill series in computer science), 414 p
53. Júnior DGS (2017) A Interação no Gerenciador de Conteúdo WordPress sob a perspectiva da Semiótica. Universidade FUMEC. http://www.fumec.br/revistas/sigc/article/view/5240
54. Japanese Society for Artificial Intelligence. Al Map Beta 2.0. Al Map Task force (2021) [cited 18 Jul 2022]. https://www.ai-gakkai.or.jp/pdf/aimap/AIMap_EN_20210901.pdf