


Article

A Platform for Integrating Internet of Things, Machine Learning, and Big Data Practicum in Electrical Engineering Curricula

Nandana Jayachandran ¹, Atef Abdrabou ^{1,*} , Naod Yamane ¹ and Anwer Al-Dulaimi ²

¹ Department of Electrical and Communication Engineering, College of Engineering, UAE University, Al-Ain P.O. Box 15551, United Arab Emirates; 700046308@uaeu.ac.ae (N.J.); 201850217@uaeu.ac.ae (N.Y.)

² College of Technical Innovations, Zayed University, Abu Dhabi P.O. Box 144534, United Arab Emirates; anwer.al-dulaimi@zu.ac.ae

* Correspondence: atef.abdrabou@uaeu.ac.ae; Tel.: +971-3-713-5149

Abstract: The integration of the Internet of Things (IoT), big data, and machine learning (ML) has pioneered a transformation across several fields. Equipping electrical engineering students to remain abreast of the dynamic technological landscape is vital. This underscores the necessity for an educational tool that can be integrated into electrical engineering curricula to offer a practical way of learning the concepts and the integration of IoT, big data, and ML. Thus, this paper offers the IoT-Edu-ML-Stream open-source platform, a graphical user interface (GUI)-based emulation software tool to help electrical engineering students design and emulate IoT-based use cases with big data analytics. The tool supports the emulation or the actual connectivity of a large number of IoT devices. The emulated devices can generate realistic correlated IoT data and stream it via the message queuing telemetry transport (MQTT) protocol to a big data platform. The tool allows students to design ML models with different algorithms for their chosen use cases and train them for decision-making based on the streamed data. Moreover, the paper proposes learning outcomes to be targeted when integrating the tool into an electrical engineering curriculum. The tool is evaluated using a comprehensive survey. The survey results show that the students gained significant knowledge about IoT concepts after using the tool, even though many of them already had prior knowledge of IoT. The results also indicate that the tool noticeably improved the students' practical skills in designing real-world use cases and helped them understand fundamental machine learning analytics with an intuitive user interface.

Keywords: IoT; AI; machine learning; big data; GUI; tool; education; engineering; curricula



Citation: Jayachandran, N.; Abdrabou, A.; Yamane, N.; Al-Dulaimi, A. A Platform for Integrating Internet of Things, Machine Learning, and Big Data Practicum in Electrical Engineering Curricula. *Computers* **2024**, *13*, 198. <https://doi.org/10.3390/computers13080198>

Academic Editor: Ananda Maiti

Received: 3 July 2024

Revised: 9 August 2024

Accepted: 12 August 2024

Published: 15 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The integration of the Internet of Things (IoT), artificial intelligence (AI), and big data analytics has revolutionized various industries, and the field of electrical engineering is no exception. Teaching electrical engineering students about the benefits of integrating these technologies is crucial for preparing them for the rapidly evolving technological landscape.

The IoT [1] has enabled the interconnection of devices and systems, allowing for seamless communication and data exchange. By incorporating IoT concepts into the curriculum, electrical engineering students can learn how to design and develop smart systems that collect and analyze data from various sources. This knowledge is essential as IoT applications continue to expand in areas such as smart grids, smart homes, and industrial automation.

AI, on the other hand, empowers machines to perform tasks that typically require human intelligence. Teaching AI to electrical engineering students equips them with the skills to develop intelligent systems capable of making autonomous decisions, optimizing energy consumption, and improving overall system efficiency. AI algorithms can analyze large datasets generated by IoT devices, enabling predictive maintenance, fault detection, and energy optimization in electrical systems.

Big data analytics complements the integration of IoT and AI by providing the tools and techniques to extract valuable insights from vast amounts of data. Electrical engineering students can learn how to process, analyze, and visualize big data to identify patterns, trends, and anomalies. This knowledge can be applied to optimize power distribution, improve energy efficiency, and enhance system reliability.

By integrating IoT, AI, and big data analytics into the curriculum, electrical engineering students can gain a comprehensive understanding of the potential benefits and challenges associated with these technologies. They also can develop the skills necessary to design and implement intelligent systems, leverage data-driven decision-making, and contribute to the development of sustainable and efficient electrical systems.

Many research works in the literature, such as [2,3], primarily concentrate on utilizing big data and AI to assist instructors in their instructional roles, whereas others focus on the individual introduction of one of these technologies in specialized courses (e.g., [1,4]). However, these approaches do not intend to provide the students with hands-on experience regarding the practical integration of IoT, big data, and ML in real-world applications, hindering their exploration of the interplay between these technological concepts.

This paper bridges this gap by developing an integrated platform that combines IoT, big data, and ML concepts. This platform is an open-source emulation software tool called IoT-Edu-ML-Stream. <https://github.com/nandanajayachandran/IoT-Edu-ML-Stream-Tool> (accessed on 11 August 2024) that can be incorporated as a part of a course or a course component of a laboratory style into an electrical engineering curriculum, enabling students to gain a deep learning experience of these technologies. The tool allows the students to create scenarios or use cases using emulated or real IoT devices that generate correlated realistic IoT sensor data in real time for various types of sensors and publish this data via the message queuing telemetry transport (MQTT) protocol [5] to a big data streaming platform, such as Apache Kafka [6]. Based on the generated data, the tool also helps the students build ML models of their choice out of different available techniques. In addition, the tool offers students near real-life experience in training, testing, and parameter-tuning their ML models.

The paper offers two main contributions. First, it introduces the design of the aforementioned emulation software tool with a graphical user interface. The tool provides practical, hands-on experience for electrical engineering students in creating IoT networks with real-time realistic IoT data generation, streaming, monitoring, and intelligent big data analytics using ML. Second, the paper provides proffered course learning outcomes that can be covered when the proposed tool is integrated into electrical engineering courses or course components. Furthermore, it introduces the reflection of these course learning outcomes on commonly used engineering program learning outcomes to help assess students' performance using an outcome-based learning approach.

The rest of the paper is organized as follows. Section 2 provides an overview of the most relevant works in the literature. Section 3 introduces the study content and significance. Section 4 details the research methodology, including research questions, the proposed tool design approach, its features, user interface, integration into electrical engineering curricula, and evaluation procedure. Section 5 presents the tool evaluation results and discussion. Finally, the paper is concluded in Section 6, highlighting future research directions.

2. Related Works

Artificial intelligence (AI) is a rapidly growing field with applications in various domains, including STEM education, big data analytics, materials informatics, and machine learning [1,4,7,8].

Several studies emphasize the integration of AI and big data into the realm of education. Khan et al. [2] propose the use of adaptive e-learning systems powered by AI and big data in higher educational institutions to assist educators and students in the teaching process and provide insights into the educational implications of these technologies in

higher education [2]. In [3], the authors discuss the design of an innovative teaching model based on AI technology, incorporating artificial intelligence hardware and software such as big data and cloud computing. The model covers pre-class, in-class, and after-class teaching, providing insights into how AI can be integrated into the teaching process [3].

Moreover, some studies describe the importance of exploring the use of learning analytics, computational analysis techniques, and AI in higher education. In [9], Montuori et al. highlight the value of learning analytics in tracking student progress, analyzing university data, and designing assessments. In [10] explores the role of human learning and learning analytics in the age of artificial intelligence, emphasizing the importance of understanding human learning processes and leveraging learning analytics to enhance educational outcomes.

Other approaches target applying AI and big data analytics in domain-specific education. For instance, in [11], the author discusses the use of big data-driven systems in advanced mathematics education. The author highlights how these systems can efficiently analyze data information, design teaching strategies, and improve teaching quality. Barenkamp et al. [12] discuss the applications of AI in classical software engineering, highlighting the automation of routine jobs, structured analysis of big data pools, and systematic evaluation using neural networks. Wei in [13] discusses the integration of information infrastructure and big data analytics in an online learning framework for rural education.

Some research works focus on enhancing employment opportunities for students via big data analytics courses. Xu et al. [14] explore the impact of digital education and the role of big data analytics courses in enhancing the skills and employability of engineering students. Tsai et al. [15] discuss precision education using statistical learning and deep learning, focusing on the impact of digital education and the role of big data analytics courses in enhancing students' skills and employability.

On the other hand, educators are recognizing the importance of incorporating AI concepts into their curricula to prepare students for the future workforce [1,4,7,8]. One approach to teaching AI is through interactive, hands-on methodologies that promote learning by doing. Xie-Li et al. [1] propose a methodology that uses open-source electronic platforms, such as Arduino, Snap Circuits, Raspberry Pi, and Circuit Playground, to teach AI concepts. Gil et al. [16] describe another way to teach AI using a workflow paradigm to capture and reuse end-to-end analytic methods, enabling students to learn about software reuse, parallel distributed programming, and high-end computing. Ayotte et al. [7] compare two different methods of teaching machine learning concepts to undergraduate students in electrical engineering. They found that a workshop approach may be more effective for student learning than a side-by-side teaching approach. Filsinger [8] discusses the design of an AI course tailored to the needs of electrical and computer engineering technology students, offering a survey of various theoretical techniques presented in a mathematically accessible way. Mannodi-Kanakthodi et al. [4] emphasizes the importance of educating materials scientists in data science and AI. They suggest providing researchers with workshops to learn and apply AI/ML tools to materials' data.

A few research papers address the introduction of IoT into electrical engineering curricula. A remote Arduino lab designed for practicing microcontroller and IoT programming is introduced in [17] for electrical and computer engineering students. Another hardware-based platform, Espressif, is proposed in [18] to develop IoT devices, enabling the students to build their own projects. In [19], Al Taai et al. cover the use of IoT in education, its role in addressing educational challenges, and its significant applications for students and teachers. It also highlights IoT's contributions to sustainable development, environmental protection, and various application areas. Some studies have introduced IoT traffic generators [20] and analyzed those data using ML without focusing on the education sector.

To harness the potential of the Internet of Things (IoT) and blockchain in education, several research studies have emphasized innovative applications and benefits. One notable study by Alammary et al. [21] conducted a systematic review of blockchain-based applications in education, highlighting the growing interest in leveraging blockchain

technology to enhance educational processes. The study provides insights into the various ways blockchain can be utilized in educational settings to improve security, transparency, and efficiency. Blockchain technology provides a secure and transparent platform for storing and sharing academic records, thereby enhancing the integrity and accessibility of educational resources, as well as streamlining administrative processes [22]. Through the use of blockchain, educators and learners can track learning outcomes, address educational challenges, and verify the authenticity of educational documents [23,24]. Furthermore, blockchain can support decentralized education systems, fostering collaboration among educational institutions and driving advancements in education quality [25].

Moreover, IoT technologies can monitor students' physical fitness data, improve supply chain traceability, and enhance logistics management within educational settings [26,27]. IoT serves as a valuable educational tool for learning programming, addressing educational issues, and offering essential applications for both students and teachers [19]. Additionally, IoT can help overcome barriers to education, such as geographical constraints and economic disparities, providing new opportunities for educational access and inclusivity [28].

Some other research works address the combination of blockchain and IoT in education. For instance, blockchain technology can enhance data security, traceability, and interoperability in IoT applications, ensuring the integrity of educational data and systems [29]. By merging blockchain and IoT, educational stakeholders can explore decentralized learning models, enhance documentation processes, and improve the overall efficiency of educational operations [30]. The adoption of blockchain and IoT in education tackles current challenges and sets the stage for future innovations and progress in the education sector [31].

In summary, the aforementioned research works mainly address two research directions. The first is the usage of IoT, big data analytics, and AI in education to facilitate the teaching process, enhance learning outcomes, and improve the employability chances of students. Additionally, this research direction explores the use of IoT technologies to monitor and enhance educational environments, as well as the application of blockchain technology to secure and optimize educational processes. The second direction is teaching AI concepts in different domains, including interactive, hands-on methodologies, intelligent workflow systems, materials informatics workshops, and tailored AI courses for specific disciplines. However, these works do not tackle preparing students with hands-on experience in the practical integration and interplay of IoT, big data, and ML in real-world applications.

Different from the reviewed research literature, the proposed tool offers a hands-on approach that enhances students' practical experience by immersing them in real-world IoT, big data, and AI environments. By working with actual hardware and software components, students gain valuable skills in data generation, network configuration, big data handling, and machine learning integration. This practical exposure prepares them for the job market, equipping them with the knowledge and confidence needed to tackle real-world challenges in the rapidly evolving fields of IoT, big data, and machine learning. Through this tool, the students will experience the transition from theoretical understanding to practical proficiency, making them highly competitive in their future careers.

To the best of our knowledge, no other educational tool has been proposed with target learning outcomes in the literature for electrical engineering students, with seamless integration of IoT, big data, and ML technologies. This integration is achieved via a GUI platform capable of real-time data generation using either real hardware or software emulation, allowing students to configure IoT devices, handle large volumes of data, and apply machine learning techniques through a user-friendly interface.

3. Study Content and Significance

3.1. Internet of Things (IoT)

The Internet of Things (IoT) is a vast network that connects everyday objects and devices, granting them the ability to observe, communicate, and collaborate effortlessly. As IoT continues to evolve, it holds the potential to transform various industries, enhancing connectivity, intelligence, and efficiency while enriching our daily experiences. At its

core, IoT relies on devices and objects equipped with embedded sensors that establish connections with one another via the Internet [32]. These devices share data with a central managing platform, which, in turn, provides insights through specialized applications tailored to specific needs. Thus, IoT can stream and automate numerous daily tasks across industries for many objectives, such as reducing costs, increasing productivity, enhancing safety, improving customer experiences, and generating new revenue streams, to name a few. In essence, IoT can empower engineers in various engineering disciplines to make more informed decisions.

As more and more devices become connected to the internet, it is becoming increasingly important for electrical engineering students to understand the principles of IoT. This knowledge can be applied to a wide range of promising industries, including, but not limited to, e-healthcare, smart transportation, and smart cities, which pave the way for students to pursue their careers after graduation in fields related to these industries. By studying IoT, students can learn about the various components that make up an IoT system, including industry-standard protocols and communication mechanisms used to connect these components, such as the MQTT protocol.

3.2. MQTT

MQTT is a lightweight and robust open messaging protocol designed for efficient IoT/machine-to-machine communication using a publish/subscribe technique [5]. It is tailored for IoT, offering secure bidirectional communication, real-time message delivery, and scalability to connect millions of devices via publish/subscribe operations even when not simultaneously connected.

When the MQTT client seeks to establish a connection with a central node called the MQTT broker, it initiates a TCP (Transmission Control Protocol) connection to the broker's designated IP address. Once connected to the broker, the MQTT client can send messages to specific topics, which serve as data transmission channels or destinations. The message can encompass a wide array of data types. MQTT imposes no limitations on the payload format, making it adaptable for diverse IoT applications. The MQTT client designates the topic to which it intends to send the message. Other MQTT clients, functioning as subscribers, can subscribe to particular topics to receive messages sent to those topics. This subscription enables them to receive messages published on a topic a client has subscribed to. Upon receiving the message, the broker forwards it to all subscribers interested in that topic. The broker ensures the delivery of that message to the subscribing client. The connection establishment diagram is shown below in Figure 1.

MQTT has developed a reputation as an efficient protocol that empowers IoT devices. Its minimal overhead lowers bandwidth usage, computing, and memory load on resource-constrained communicating nodes, such as IoT devices. MQTT achieves efficient data transmission in large-scale IoT networks through its lightweight nature, which minimizes the communication network traffic [5]. MQTT is currently a common choice for applications such as remote monitoring, patient health tracking, and danger alerts. Its efficient and scalable nature, combined with the publish/subscribe model and support for real-time communication, makes MQTT popular in the industry and, hence, a good choice for the proposed platform to help students gain valuable practical experience.

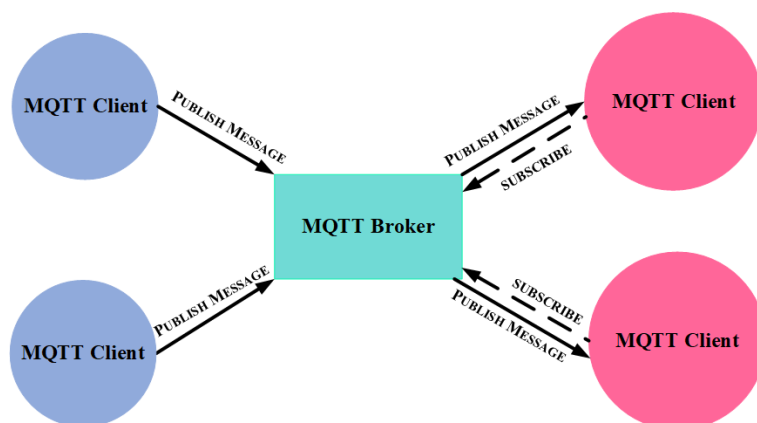


Figure 1. MQTT connection establishment.

3.3. Big Data and Big Data Platforms

The widespread adoption of technologies like IoT, social media platforms, and the significant increase in digitization across various sectors has led to the generation of massive amounts of data, commonly referred to as “big data”. Big data encompasses vast and complex datasets that exceed the capacity of traditional data processing tools in terms of capturing, storing, managing, and analyzing [16]. It is characterized by three primary attributes: volume, velocity, and variety. Volume pertains to the sheer scale of available data, velocity refers to the speed at which data accumulates, and variety encompasses the diverse data sources. Additionally, two more attributes are commonly associated with big data: veracity (consistency and reliability of data) and value (usefulness of insights extracted from the data).

Big data platforms are specifically designed to handle large, dynamically generated, complex datasets in real time. These tools, developed by data management vendors, play a pivotal role in enhancing the scalability, accessibility, performance, and security of organizations relying on big data analytics. In today’s landscape, a wide range of big data tools is available, spanning data stores, development platforms, integration tools for reporting, and advanced analytics. Prominent examples include Hadoop, Apache Spark, Apache Kafka, Cassandra, and Rapid Miner [14].

Notably, Apache Kafka, a renowned open-source platform, has emerged as a favored choice within the realm of big data, enabling real-time data streaming and processing at scale. It is a powerful and extensible open-source event streaming platform designed to facilitate the creation of real-time data pipelines and streaming applications [6]. It utilizes a publish–subscribe messaging system that excels at rapid and robust data processing. One of Kafka’s standout features lies in its complete decoupling of data producers and data consumers, offering unmatched scalability. Kafka’s architecture revolves around the concept of topics, which act as data containers accommodating multiple producers and consumers/subscribers, facilitating the repeated reading and processing of data/events. Importantly, Kafka’s performance remains consistent regardless of data volume, making it an ideal choice for long-term data retention.

3.4. Machine Learning

Machine learning is a field that involves the development and understanding of models that have the ability to learn from data. ML algorithms enable models to approximate any function when provided with sufficient data in the right format. These algorithms fine tune themselves based on the data they are trained on. Thus, it is a sort of artificial intelligence that allows computer systems to learn and develop without being explicitly programmed. It entails using algorithms and statistical models to analyze data, detect trends, and make predictions or judgments based on the data without human intervention. ML popularity is fueled by the advances in chip computation, the abundance of data we

can now collect, and the continuous increase in computational power required for machine learning algorithms to run efficiently.

Teaching ML to electrical engineering students is important due to the current increasing demand for intelligent systems and the need to analyze vast amounts of data collected in real time from IoT devices. This makes machine learning a critical component of many modern technologies, including self-driving cars, voice assistants, predictive maintenance systems, and many others. Therefore, teaching ML to these students is essential for preparing them for the future of technology and ensuring that they have the skills needed to succeed in a rapidly changing job market.

3.5. Integration of IoT, a Big Data Platform and ML

The integration of a big data platform with IoT and machine learning not only enhances data processing capabilities but also facilitates real-time actions based on intelligent insights. Figure 2 shows a simple integration of the domains. With a big data platform, e.g., Kafka, providing a robust data transport mechanism, organizations can efficiently manage the influx of data from diverse IoT sources, ensuring data integrity and timely delivery. Machine learning models integrated into this pipeline can analyze data for patterns, trends, and anomalies in real time. For example, in a multidisciplinary application of agriculture and electrical engineering, wireless sensors in farms can transmit environmental data to Kafka, where machine learning models can provide insights into optimal irrigation schedules or pest control strategies. This synergy creates a closed-loop system in which insights generated by machine learning algorithms can trigger immediate actions, such as adjusting settings on connected devices and automating decision-making processes. The integration of Kafka, IoT, and machine learning opens doors to a wide array of applications that drive efficiency, cost savings, and innovation across industries.

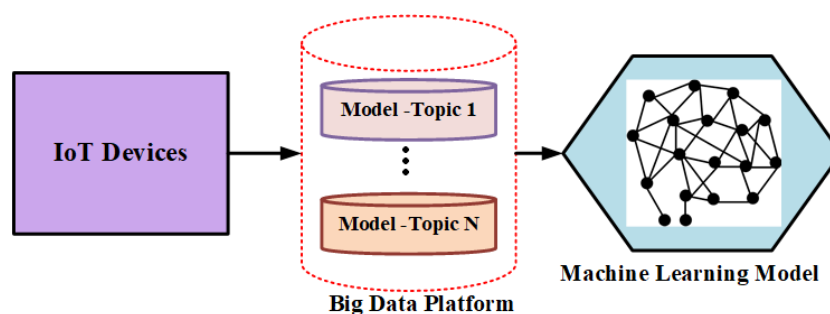


Figure 2. Integration of IoT, big data platform, and ML.

Students who gain hands-on training in these technologies will be at the forefront of innovation, leading the charge in industries like e-healthcare, smart agriculture, smart transportation, and smart cities. Practical knowledge of these technologies is crucial, as it equips students with the skills required to work in the real world. Training allows students to understand the complexities of these technologies, including the protocols and communication mechanisms used to connect various components. Furthermore, practical training helps students develop analytical and problem-solving skills. By working with real data sets, students learn to analyze and interpret data, identify patterns and trends, and derive insights that can be used to drive decision-making processes. This knowledge prepares graduates to make informed decisions, ultimately driving innovation and efficiency across many different industries.

4. Research Methodology

This section introduces the research questions, the tool design, and the integration of the tool into the curriculum.

4.1. Research Questions

This research mainly tackles the following research question: How can we design an educational tool that provides electrical engineering students with hands-on experience in the integration of IoT with big data analytics using basic ML techniques? Another related question is: How can such a tool be integrated into typical electrical engineering curricula?

4.2. Tool Design Approach and Features

Our main design objective is to create a GUI-based tool that integrates IoT, big data, and machine learning (ML) to provide hands-on experience for electrical engineering students in these fields. Consequently, the tool design should equip the students with hands-on experience in emulating IoT devices using the MQTT protocol, introduce them to real-time big data platforms, and encourage them to explore and analyze different ML techniques in a user-friendly manner. Therefore, the tool shall have an iterative structure that advocates correcting errors and developing proficiency through the educational journey.

To achieve the tool design objective, we followed a GUI paradigm that enables the students to interact with the tool via a step-by-step option-based approach. This approach guides them first to set up either real hardware or synthesized (virtual) IoT devices by selecting the types of devices, adding sensors to them, and configuring them as MQTT clients. A packet capturing tool (Pyshark [33]) is used to obtain the IoT data generated in real time either from hardware or data synthesis. Afterward, the data are streamed to a big data platform (Kafka). The streamed data are subsequently utilized to conduct machine learning analysis based on the machine learning algorithm chosen by the student user from a list of available algorithm options.

Furthermore, the tool design exemplifies scalability, effectively handling substantial volumes of IoT data. The tool is exclusively developed using Python, the industry standard for AI development. It can be further customized and extended to meet specific use cases and requirements, offering flexibility to adapt its functionality as needed. Figure 3 highlights the design features of the tool.

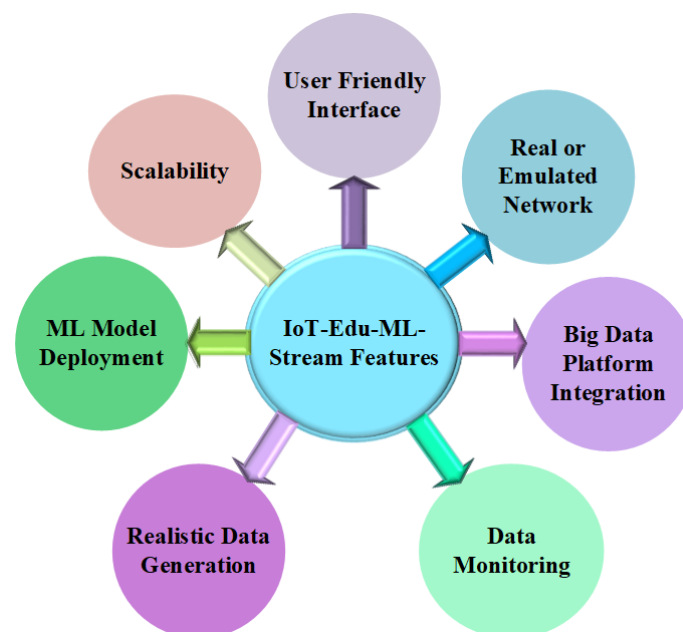


Figure 3. IoT-Edu-ML-Stream features.

Since the tool usage covers the interplay between IoT, big data, and machine learning, the tool design structure is modular, as illustrated in Figure 4. Initially, an IoT data generation module generates IoT data, which is subsequently captured, processed, and sent to the big data module that streams data in real time to the big data platform (Kafka). The

real-time streaming data are then used for ML data analytics through the chosen machine learning module. The design and functionality of each module are detailed in the sequel.

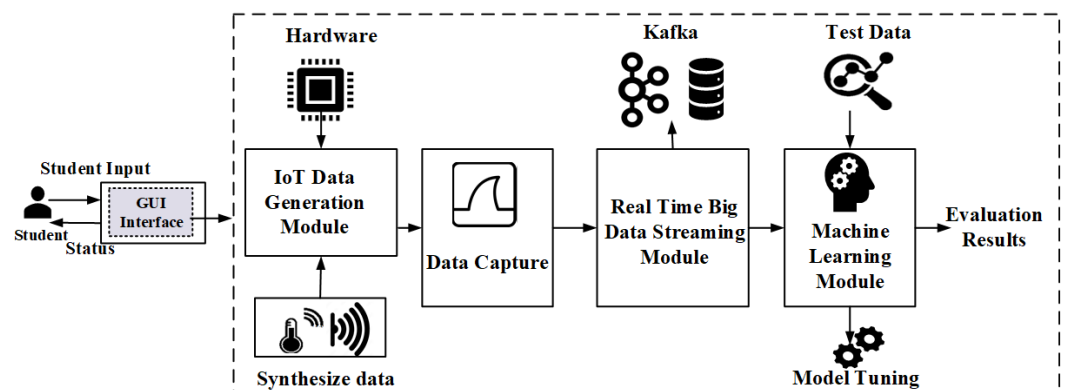


Figure 4. Design approach.

4.2.1. IoT Data Generation Module

This module within the tool emulates IoT devices or interacts with real ones. The output of this module is MQTT network packets that contain realistic and correlated sensor readings. Initially, students set up essential parameters in this module, enabling the creation of IoT data from one of two options, namely, real hardware devices or emulated (synthesized) sensors of different types (e.g., temperature, humidity, air pressure, etc.) with realistic and correlated readings based on widely available datasets. In this module, student users can define the IoT network devices of their designed use cases by adding devices and attaching sensors to them. Currently, the tool supports only the MQTT protocol for transferring the sensed data.

Clients and MQTT Broker

The IoT devices take on the role of MQTT clients. These clients are responsible for sending and receiving data within the IoT network. The key component in MQTT communication is the MQTT broker, which serves as an intermediary, ensuring seamless communication between MQTT clients. It plays a pivotal role in routing messages from publishers to subscribers, facilitating efficient data exchange.

IoT Network Configuration

Students can create use cases, add and configure devices, and specify the number of devices and sensors per device. These configurations collectively form the IoT network configuration.

Data Generation Profile

Students can set up the data generation profile for periodic or random data generation and define periods or time intervals for synthesized data generation. This configuration effectively emulates real-time sensor data. For generating data from real hardware, the hardware sensor nodes should be configured to follow the same profiles.

Data Generation

For the hardware option of the tool, a physical setup can be used to generate real-time IoT data, an example of which is shown in the tool evaluation Section 4.4. On the other hand, for the synthesizing option, we developed an IoT data generator that utilizes pre-trained machine learning models. These pre-trained models are capable of generating correlated sensor readings. We have included temperature, humidity, motion, and light sensors in the current data generation module. Moreover, the synthesizer is customizable to generate other sensor readings whenever required. It uses a Random Forest-based pre-trained model to generate correlated data and the MQTT protocol for data publishing. The pre-trained ML model is developed using a combination of real-time open-source datasets [34] and hardware-synthesized data. It operates by loading the pre-trained model and retrieving configuration settings provided by the students, which include the MQTT

topics, IP addresses for the clients and broker, and the data publishing frequency. Based on the network configurations, the pre-trained model predicts correlated sensor readings and publishes them to the specified MQTT topics. This approach allows students to emulate IoT data generation scenarios or use real hardware to generate them, providing a comprehensive learning experience.

4.2.2. Real-Time Streaming Module

The real-time streaming module is responsible for capturing the IoT data generated by the previous module and streaming it to a Kafka-based big data platform. The module operates in the following stages:

Data Capture

In this stage, PyShark is used as a Python module designed to serve as a bridge between Tshark and Python, enabling parsing network packets in Python by utilizing Wireshark dissectors [33].

Big Data Platform Integration

Captured data are seamlessly transferred to a real-time streaming big data platform (Kafka) in this stage. The Kafka processes involve establishing a cluster, creating meaningful topics to categorize and classify data logically, and then aligning with the IoT data generation module by naming topics after device names and sensor types. Integration with Kafka is accomplished through well-defined producer and consumer configurations [6].

The Kafka producer is set up with settings such as idempotence, ensuring that messages are delivered exactly once without duplicates. The compression type is set to "lz4", reducing data size to increase transmission efficiency. Other settings include batch size, defining the amount of data sent in one go to optimize throughput, retries to handle message resend attempts in case of failures, and delivery timeouts to ensure messages are sent on time.

For the consumer, configurations include settings to read data from Kafka topics and perform pre-processing steps, such as removing redundancy and converting byte data into a readable format before forwarding it to the appropriate Kafka topics. Additional settings involve isolation levels, offset management, and auto-commit configurations. Isolation levels ensure data integrity by specifying how transactions are handled. Offset management tracks the reading progress within each topic, enabling the consumer to resume from the correct position. Auto-commit settings streamline read position updates by automatically recording the read progress. These setups ensure the correct formatting and readiness for further use. In essence, this integration involves the creation of model topics within Kafka to manage and organize incoming data efficiently. This real-time streaming process should be continuous and uninterrupted.

4.2.3. Machine Learning Module

The last module involves machine learning analysis of the real-time streamed data, where students can either load a dataset already created or generate the dataset in real time. The ML module is capable of handling both regression and classification tasks. The main steps involved are described below.

Data Preprocessing and Dataset Splitting:

After loading the dataset, the student has the option to perform data preprocessing. This step involves cleaning and transforming the data to ensure it is suitable for machine learning. Preprocessing includes tasks such as handling missing values and outliers, scaling features, and encoding categorical variables if necessary.

Since students can generate datasets in real time, they have the option to specify the number of data records they want. There is no need for explicit programming from the students. Once the data are preprocessed, the student is prompted to split the dataset into training and testing subsets. This step is crucial for assessing the ML model's performance. Typically, a portion of the data is allocated for training, while the remaining portion is

reserved for testing. This separation allows the model to learn patterns from the training data and evaluate its generalization on unseen data.

Regression and Classification:

The current machine learning analysis available in the IoT-Edu-ML-Stream tool encompasses both regression and classification tasks, depending on the nature of the use case. For regression, the goal is to predict continuous output variables based on input features, such as predicting temperature from sensor readings. The tool uses typical evaluation metrics for regression, such as mean squared error (MSE) and R-squared, which assess prediction accuracy. The tool can be extended for classification tasks to include models that assign data points to predefined categories. Metrics such as accuracy, precision, recall, and F1 score can be used to evaluate classification performance, helping users understand the model's effectiveness in categorizing data.

Model Fitting and Optimization

Once the machine learning model is chosen and configured, it is fitted to the training data. Model fitting involves adjusting the model's parameters to find the best-fitting model for the given data. The iterative training process explores various combinations of parameters and configurations to optimize model performance.

The tool currently contains four model options available: a linear model (linear regression or logistic regression, depending on the task), a tree-based model (Random Forest), an ensemble method (Gradient Boosting), and a neural network model. This setup is customizable, allowing the future addition of any number of supervised algorithms. The trained machine learning models are then saved for future predictions within the same tool session. These trained models can make predictions based on the input data obtained from the model topic, which could include real-time IoT sensor readings. Such predictions are crucial for identifying patterns and correlations among various inputs, such as temperature and humidity. Students can gain insights into how different factors influence the output variables by analyzing the model's predictions, helping them understand and better tune the IoT data generation module parameters.

4.2.4. Operation Procedure

The flowchart in Figure 5 summarizes the flow through the tool operation procedure. As the figure shows, when initiating the IoT-Edu-ML-Stream tool, students will have two choices: start a new project or load an existing one. Next, they can either generate data from hardware devices or synthesize it. In either case, they should configure the IoT network by adding devices and selecting the required sensors. Once the devices and the MQTT broker are configured, they will create big data topics (according to the devices and sensors they added). This step will also be reached if they choose to load an existing project at the beginning. After that, they start the IoT data generation. Following data generation, the generated IoT data are captured using a data capturing module employing PyShark. Students can view the big data topics and their contents as needed.

Subsequently, the ML module comes into play. Students are supposed to begin by configuring the ML model, starting with the option to select the desired time limit for loading the data being streamed. Since the data are streamed in real time, setting a longer time limit for data generation guarantees the availability of a sufficient amount of data in the corresponding Kafka topics. After loading the dataset, student users can specify the ML model inputs from the big data platform topics. If necessary, data preprocessing can be performed at this stage. Since the streamed data are in real time, there might be missing values in the dataset, which should be treated before fitting the model. Here, the students save the dataset and are provided with options to view it. If they find the need for preprocessing, they can do it; otherwise, they can skip this step. According to the ML model selected, students will have to adjust the parameters for the model in the ML tuning step. For each selected model, the parameters to be tuned will be different.

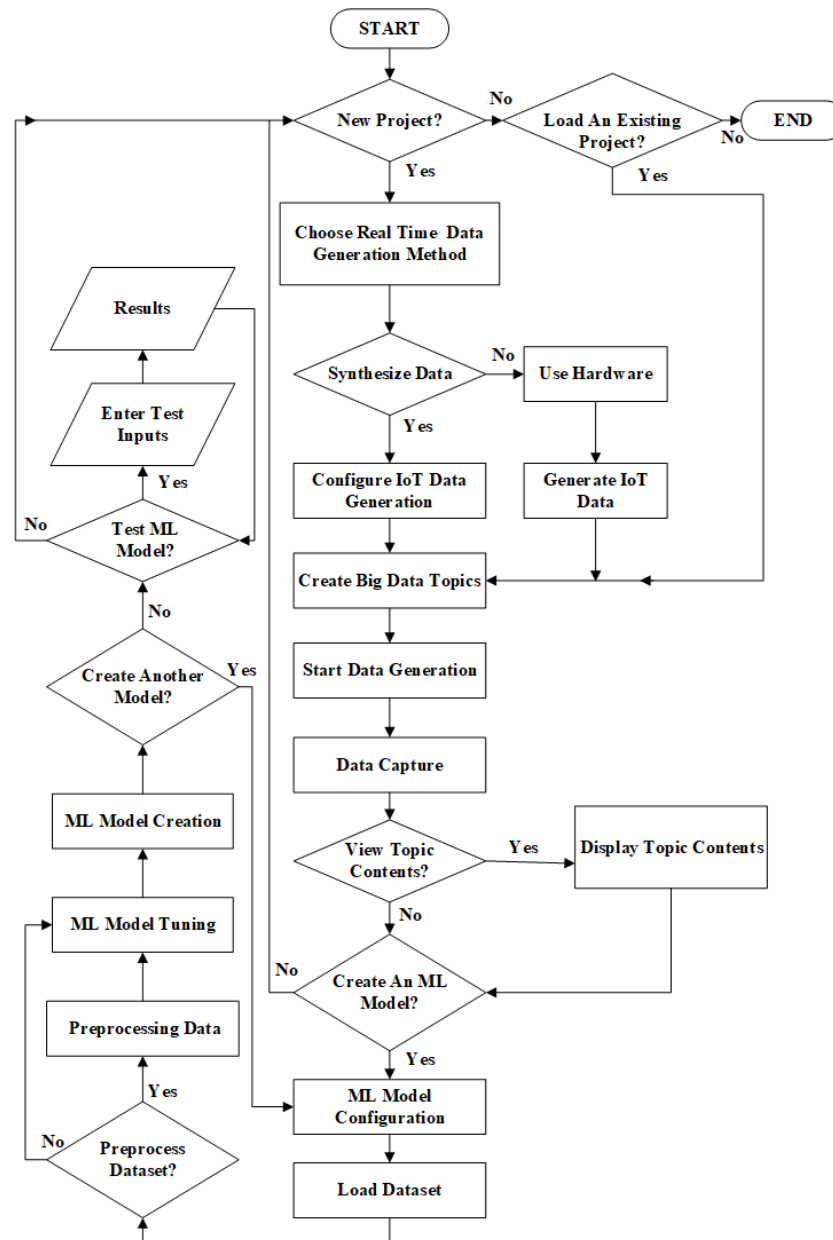


Figure 5. IoT-Edu-ML-Stream flowchart.

Once an ML model is created, students are prompted to decide whether to create another one. If they choose to do so, they will be directed to the ML model creation step again. Otherwise, they can proceed to test the ML model by adding user inputs and viewing the results. If they choose not to test the model, they will be redirected to the new project creation menu, where they have the option to exit the tool.

4.2.5. The Tool User Interface

The IoT-Edu-ML-Stream GUI tool offers a user-friendly interface for setting up IoT data generation, real-time streaming, and ML analysis of the streamed data. The screens are designed with help text added to guide students through the concepts and ensure smooth navigation. The process begins with the students launching the GUI of the tool and selecting whether to generate real-time data from hardware or synthesize data in real time through emulation, as shown in Figure 6. This initial choice determines the data source for their IoT network configuration.

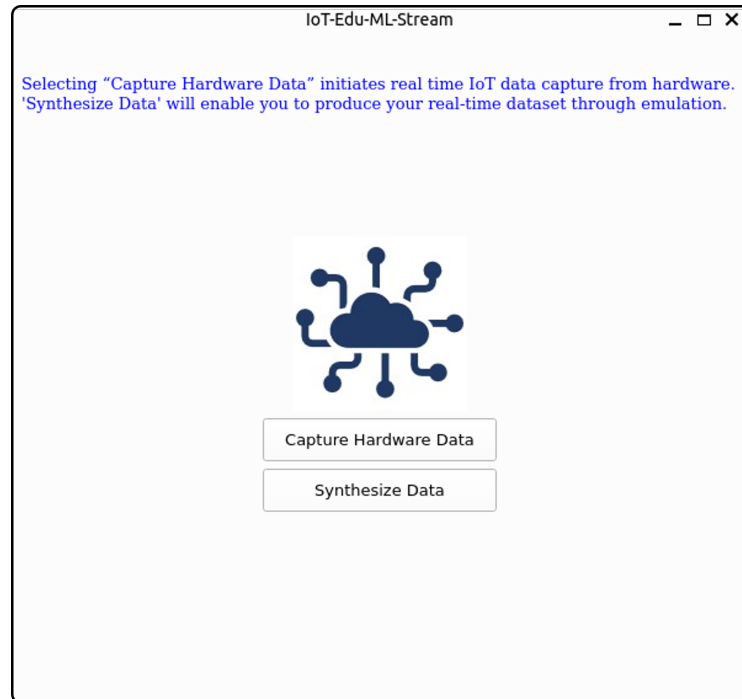


Figure 6. Screen to select data generation method.

After selecting the data generation method, students create their IoT network by selecting devices and adding sensors, as revealed in the screenshot of Figure 7.

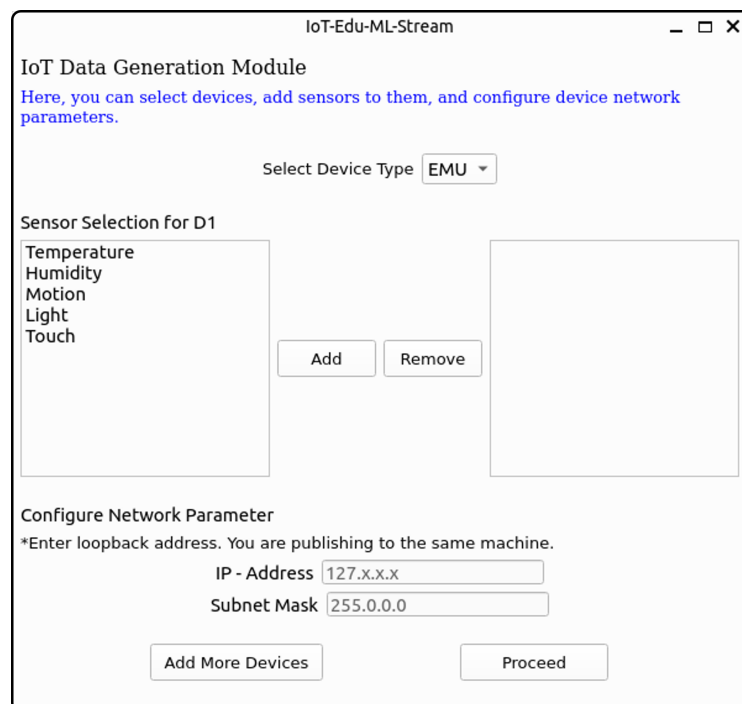


Figure 7. Screen to create IoT network.

Students will then configure the network, which consists of the IoT devices (acting as MQTT clients) and the MQTT broker, by entering the IP addresses and network masks of all the network entities. As shown in Figure 8, loopback addresses are used for both the broker and the client as they communicate within the same computer in the synthesized data scenario. The screen in Figure 9 provides a summary of the selected devices and IoT net-

work configuration. Clicking “Proceed” initiates data emulation, enabling communication between the MQTT broker and the defined clients.

Figure 8. Screen for IoT network configuration.

Device Name	Device Type	Sensors	IP Address
1 D1		Temperatu...	127.0.0.2
2 D2		Temperatu...	127.0.0.5
3			
4			

Broker Type	IP Address
1 MQTT Broker	127.0.0.1

Type	Range (s)
1 Random	5

Figure 9. Screen showing network configuration summary.

Data streamed from the IoT network are stored in the big data platform (Kafka) under specific topics. The screenshot in Figure 10 shows the topic names to be created and includes an option to start the data streaming by creating the big data topics. The nomenclature used here is as follows: for “Device 1” with the sensor “Temperature”, the topic name will be written as “D1-Temperature”.

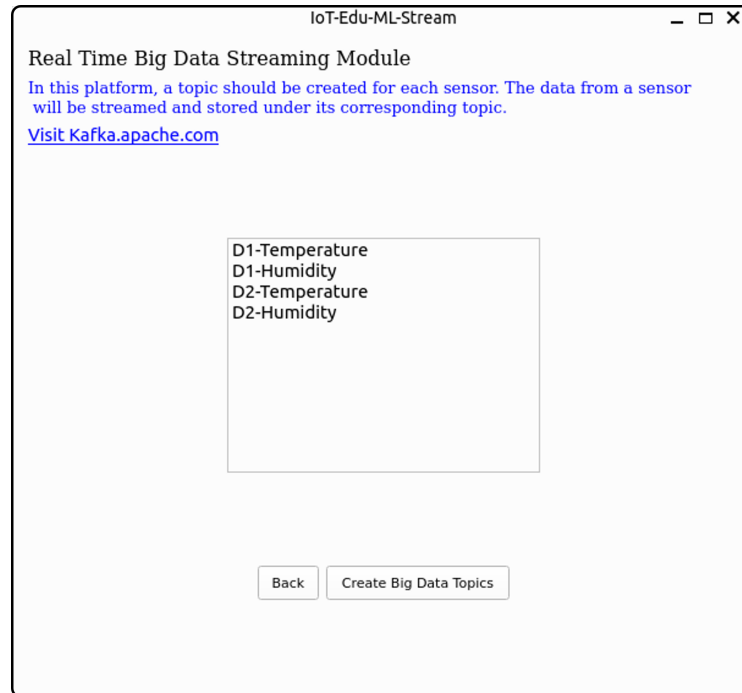


Figure 10. Screen to create big data topics.

Students begin building their machine learning models by defining the input data. They create datasets for ML tasks from the data stored in the big data platform. The process of selecting the input features for the model is shown in Figure 11.

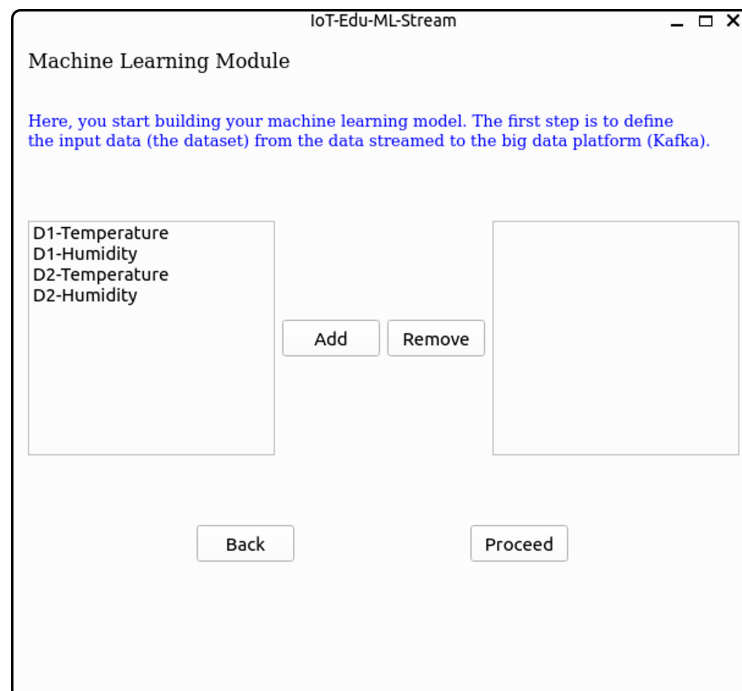


Figure 11. Screen for choosing ML input data.

Student users are then allowed to save the prepared dataset in a comma-separated value (CSV) format, as shown in the exhibit of Figure 12 by specifying the number of data records needed in the dataset file.

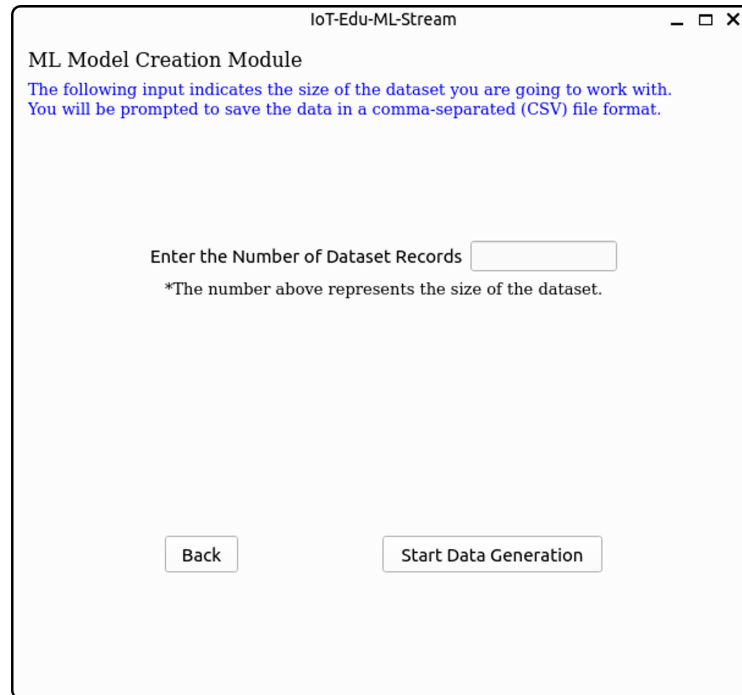


Figure 12. Option to save the dataset in CSV format.

This CSV file can then be uploaded for ML tasks. After uploading the CSV file, students can select a suitable machine learning algorithm to build their model, as revealed in Figure 13. Each algorithm is accompanied by a brief description to aid in selection, guiding students in choosing the appropriate technique for their task.

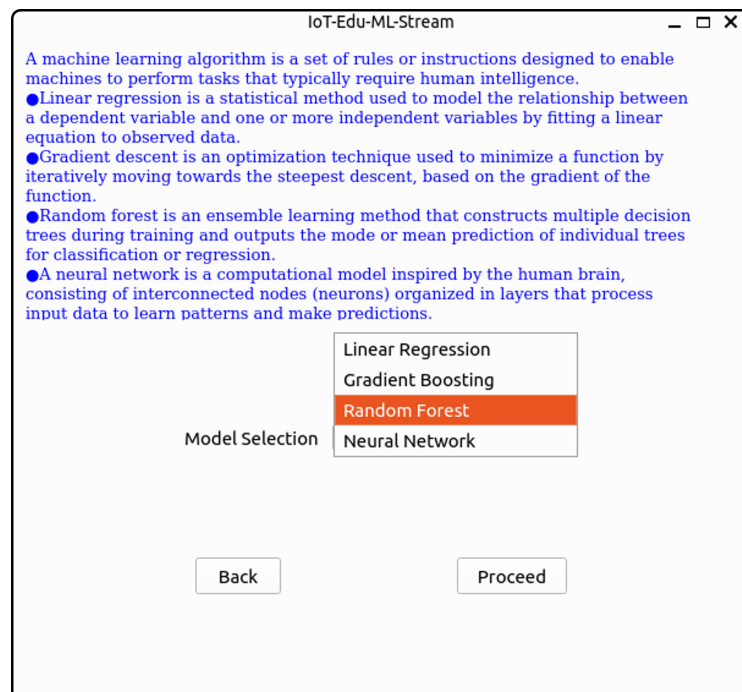


Figure 13. Screen to choose ML algorithm.

The screenshot in Figure 14 shows how the students can configure the parameters of their chosen machine learning algorithm. For instance, if the students select the linear regression (LR) model, they can tune parameters such as whether to fit the intercept and normalize the data. If they choose the neural network (NN) model, they will adjust param-

eters such as the number of layers, neurons per layer, and the optimization algorithm (e.g., Adam, SGD, or RMSprop). In this instance, they are tuning the parameters of the Random Forest model, where they can set the number of estimators and the maximum depth of the trees. Once the parameters are set, clicking “Train” initiates the model training process, allowing the model to learn from the training data based on the specified configurations.

Figure 14. Configuration of the parameters.

The tool later displays the evaluation metrics for the trained model, as shown in Figure 15. After this step, students can choose to test the created model, train the model with a different algorithm, or use a different dataset for training the model. This screen allows them to evaluate and refine their machine learning models.

Accuracy	:	0.85
MSE	:	0.2
F1 Score	:	0.8

Figure 15. Model metrics and options.

4.3. Tool Integration in Electrical Engineering Curriculum

The IoT-Edu-ML-Stream tool can benefit electrical engineering students if introduced to support a general course about IoT, ML, and big data. It can also be introduced as a component in a senior-level specialized course in the electrical engineering curriculum addressing modern application use cases that require IoT sensing and communication with intelligent big data analytics, such as courses about industrial automation and smart power grids.

This is because a specialized course or course component for electrical engineering students that integrates ML, IoT, and big data should provide an in-depth understanding of the fundamental concepts of AI and machine learning. Students are supposed to gain knowledge and skills related to various machine learning techniques, such as supervised learning. This understanding is essential for designing and implementing the use cases addressed by the course subject. Moreover, students shall also learn about the basic principles of IoT and how it relates to AI. This requires covering topics related to data acquisition, sensor/device networking, and data processing, which are essential for building intelligent systems interacting with the physical world. Furthermore, the students need to learn how to manage and analyze large amounts of data, which is essential for making decisions based on data-driven insights.

By using the proposed tool, students are supposed to gain the skills and knowledge to design and implement ML, IoT, and big data use cases. They will be able to analyze data using various ML techniques and evaluate the performance of these techniques.

In the following, we introduce proposed course learning outcomes for integrating the tool into an electric engineering curriculum and outline the benefits of this integration and some limitations.

4.3.1. Proposed Learning Outcomes for Curriculum Integration

Following an outcome-based learning approach, some suggested course learning outcomes (CLOs) that can be targeted while using the IoT-Edu-ML-Stream tool in a specialized course or course components are introduced in the following.

After using the tool and by the end of studying the required material, students should be able to:

- CLO1: Outline the fundamental concepts of AI and machine learning.
- CLO2: Identify the basic principles of IoT and how they relate to AI, including data acquisition, processing, and analytics.
- CLO3: Recognize big data platforms and related big data technologies.
- CLO4: Design an IoT-based application use case involving intelligent decision-making via data analytics.
- CLO5: Compare the performance of different ML techniques.

Since many engineering programs all over the world are accredited by the Engineering Accreditation Committee of the Accreditation Board for Engineering and Technology (ABET), we introduce here some student outcomes or program learning outcomes (PLOs), following the ABET guidelines (with the same numbering of the outcomes) [35]. These PLOs (seven) have been proposed by ABET as general student learning outcomes that can be targeted by any engineering program. These PLOs define the specific knowledge, skills, and behaviors the students are expected to possess. These outcomes are indicative of the progression students make throughout the program, reflecting the acquisition of essential skills, knowledge, and behaviors. These PLOs can be addressed by including the proposed tool in an electrical engineering curriculum. Table 1 outlines the mapping between the suggested CLOs and four ABET engineering PLOs. This mapping shows some of the PLOs (defined by the ABET number and description in the first two columns) that can be covered by the proposed CLOs and the most relevant CLOs (last column) to each. This implies that the attainment of relevant CLOs refers to the attainment of their corresponding PLO. The table shows that the suggested CLOs can map to more than half of the program outcomes

set by ABET guidelines (seven) [35], indicating the proposed tool's effective contribution to electrical engineering programs.

Table 1. The ABET program learning outcomes [35] addressed by the usage of IoT-Edu-ML-Stream mapped to the suggested CLOs.

PLOs	Description	CLO Mapping
First	<i>"An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics"</i>	CLO-2,4,5
Second	<i>"An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors"</i>	CLO-4,5
Sixth	<i>"An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions"</i>	CLO-4,5
Seventh	<i>"An ability to acquire and apply new knowledge as needed, using appropriate learning strategies"</i>	CLO-1,2,3

4.3.2. Integration Benefits

Integrating the IoT-Edu-ML-Stream platform into the curriculum provides several key benefits. Students gain a comprehensive understanding of the combined effects of IoT, big data, and machine learning. They learn essential IoT concepts, including network configuration, client–broker communication, and IP address setup, through both emulation and hardware options. The tool also exposes students to real-time big data platforms, including data streaming and topic creation, offering hands-on experience with real-time data processing. In the ML module, students generate datasets, learn the importance of data in machine learning, and practice splitting datasets into input features and target variables. They engage in training models, understand the differences between regression and classification, and explore preprocessing and parameter tuning. The tool guides students through evaluating and testing models, providing a complete ML process experience. Additionally, the tool's user interface offers instructions, helpful hints, and links to further resources, enhancing student engagement and encouraging deeper learning.

4.3.3. Limitations

The current version of the tool is designed to handle only the MQTT protocol, which may limit its conformance with other IoT communication protocols. The tool setup is a one-time process but requires expertise in the Linux operating system to ensure the needed software packages are properly installed. The hardware devices, if required to generate data, should be suitably pre-configured to ensure data communication is working as it should. Additionally, the machine learning module within the tool is currently limited to handling only supervised learning techniques, restricting the scope of ML applications that can be explored using the platform.

4.4. Tool Evaluation

To assess the effectiveness of the IoT-Edu-ML-Stream platform, we conducted an evaluation survey among senior undergraduate electrical engineering students, formatted as a five-point Likert questionnaire. The survey consists of six questions categorized into three sections. In the first section of the survey, respondents rated their prior knowledge related to IoT, ML, and big data. The questions in this section are meant to analyze the participant's understanding of IoT with big data and machine learning concepts after using the tool to evaluate the knowledge gained compared to prior knowledge. The second section focuses on the effectiveness of designing application use cases with the tool. In the third section, we evaluated the feedback on how well the tool introduced real-world scenarios. By organizing the survey this way, we aimed to cover all aspects comprehensively. The survey questions are listed as follows:

- Q1—Rate your prior knowledge in the integrated domains of IoT (Internet of Things), big data, and ML (Machine Learning).

- Q2—Rate your knowledge gained about IoT concepts after using the tool.
- Q3—Rate to what extent you grasped the basic concepts of machine learning using the tool.
- Q4—Rate the level of difficulty in designing use cases using the tool.
- Q5—Rate how much the tool enhanced your practical skills for real-world applications.
- Q6—Rate how user-friendly the tool's interface was.

4.4.1. Participants and Data Collection

The survey involved the participation of 29 senior undergraduate students with different background levels in IoT and ML. The students were exposed to the proposed platform during the activity mentioned in Section 4.4.2. After they finished the activity, the provided survey responses were collected from the survey participants, who were informed about the purpose of the survey and were assured of the confidentiality of their responses. The survey was distributed using the secure and user-friendly platform SurveyMonkey, ensuring that participants could easily provide their responses anonymously. Each participant was granted access to the proposed platform for a specific period.

4.4.2. Tool Evaluation Activity

During this period, the participating students actively interacted with the tool's intuitive graphical user interface (GUI) to gain practical, real-world experience in IoT, big data analytics, and machine learning (ML). They were asked to design a practical use case for controlling the temperature adjustment of a room air conditioner (AC) without human intervention using ML. Figure 16 effectively illustrates the hardware setup and software packages required to implement the use case efficiently. The central component is a Linux box running Ubuntu 20.04, which operates all the essential software packages, including Python 3.10.12, Kafka 3.6.0, Pyshark 0.6, and the MQTT protocol implemented with Paho-MQTT (Mosquitto version 2.0.11). A desktop display screen was connected to the Linux box to display the IoT-Edu-ML-Stream interface. The Linux box is connected to a WiFi router, which serves as an access point for the entire system. Device 1 is a Raspberry Pi connected to a DHT-11 Humidity sensor, whereas Device 2 is an ESP-32 connected to a DS18B20 digital temperature sensor. Both devices are connected wirelessly to the router for network communication.

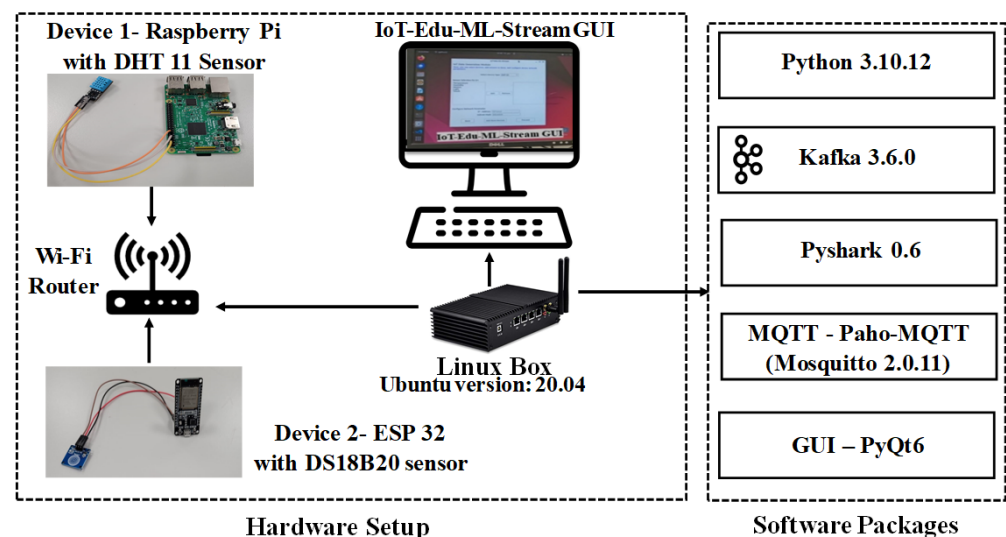


Figure 16. Block diagram outlining the required hardware and software setup for the case study.

Students were provided with the pre-configured hardware devices shown in Figure 16 to understand the generation of IoT data through hardware. They generated data by setting all the necessary configurations with the tool, thus enabling the streaming of IoT

data from humidity and temperature sensors indoors and outdoors. Figure 16 showcases the indoor hardware setup. The students were also asked to generate more data using the data synthesizer option to have a dataset of a sufficient number of data records. They emulated inside and outside devices equipped with humidity and temperature sensors, configured them using the GUI tool, synthesized real-time data, and streamed it through the big data platform.

In the ML phase, the students entered the number of data points they wanted in their dataset, captured real-time data, and saved it in a dataset. Students then labeled and uploaded the dataset and performed preprocessing. They selected the ML model they wanted to train, including linear regression, Gradient Boosting, Random Forest, and neural network models. After selecting the model, the students tuned the parameters for the chosen model and followed the explanations on the screen to proceed with training the ML model. Since they carried out the classification, they evaluated the trained model by calculating metrics such as accuracy and F1 score. The students built all the models, compared the accuracy of each, and increased the data records to see the improvement in accuracy. With the best model, they made decisions on the intelligent automated adjustment of the room AC. This hands-on activity allowed them to experience the entire process, from data generation to the practical application of ML algorithms for decision-making.

The students were asked to answer the survey questions after finishing the activity. Note that Q1, Q2, Q3, and Q5 are used to measure the student's perception of prior and gained knowledge and practical skills after completing the activity. Suppose this activity is a part of some course in an electrical engineering curriculum. In that case, the survey results can also be used as a measure of the student's perception regarding the attainment of the proposed CLOs using Table 2 (i.e., the question that receives a high average rating indicates its related CLO(s) are attained from the student's perspective). Attaining the CLOs is an essential measure of the success of integrating the proposed platform into an electrical engineering curriculum.

Table 2. Knowledge and skills survey questions mapped to the suggested CLOs.

Survey Question	Related CLO(s)
Q1	CLO-1
Q2	CLO-2
Q3	CLO-2, CLO-5
Q5	CLO-3, CLO-4

It is worth noting that the proposed platform can be applied to any scenario involving IoT sensor data. Beyond temperature control, it can be used in various use cases. For instance, in computer network security applications, students could configure IoT devices or generate emulated data for different applications. These devices can stream real-time data by properly configuring the IoT networks and training the ML models to detect unusual patterns in the reported data in order to characterize data integrity security threats. In another example, within the context of electric grids, the tool can synthesize data obtained from various sensors installed on vital equipment, such as transformers, or receive this data live via a configured IoT network from a real hardware version of the equipment installed in a university laboratory setup. These data can then be processed to build ML models to predict and identify faults in this transformer, facilitating predictive maintenance of electric grids. These examples highlight how the tool offers practical, hands-on experience across various real-world scenarios of significant importance to electrical engineering students' future careers.

5. Tool Evaluation Results and Discussion

This section presents the survey results and provides a discussion of the findings.

5.1. Tool Evaluation Results

The analysis of the survey results is presented in the bar charts below. The survey started by addressing students' prior knowledge related to IoT, big data fundamentals, and ML. The results indicated a diverse range of responses, as shown in Figure 17. The mean rating of the responses is 3.62, and the standard deviation (SD) is 1.25, indicating the variability in students' prior knowledge. About 62% of the participating students had a reasonably good understanding (rated 5 and 4). Another substantial portion perceived their knowledge to be average, while a smaller portion of about 10% indicated limited knowledge.

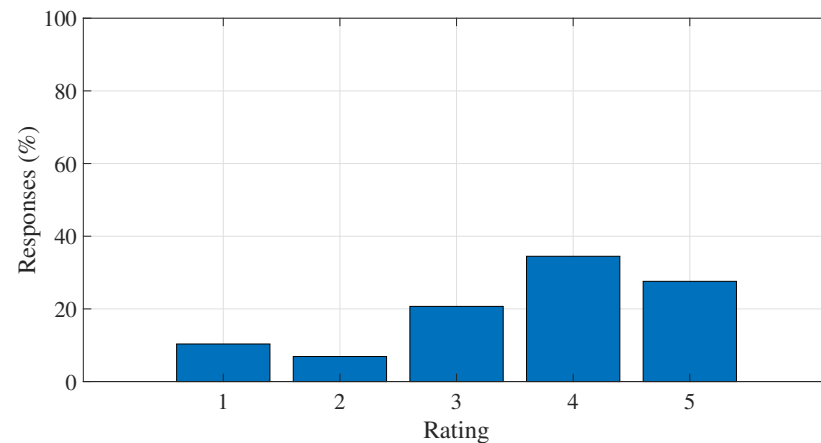


Figure 17. Q1 survey response.

Regarding developing their IoT knowledge after using the tool, Figure 18 shows that approximately 65.5% of students reported significant improvement (rated 5) in the fundamentals of these topics. In comparison, around 35% indicated very good knowledge gain. No student gave a rating below 4. The high mean rating of 4.66 and low standard deviation of 0.425 highlight the tool's effectiveness in enhancing IoT knowledge.

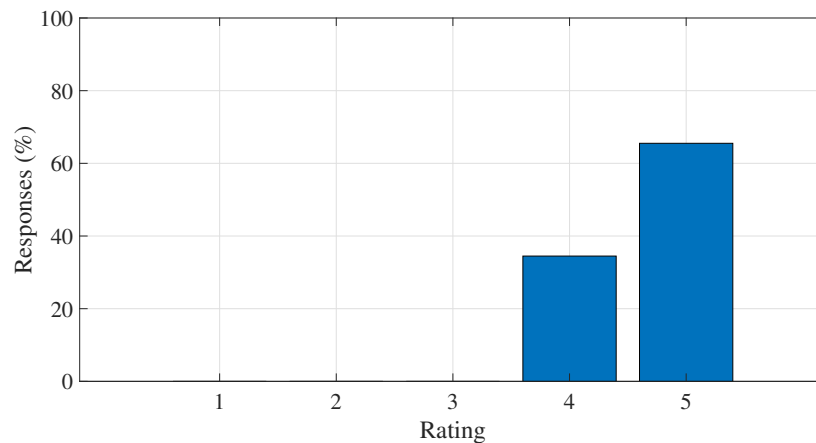


Figure 18. Q2 survey response.

When asked the participants how they grasped ML concepts conveyed by the tool, about 93.1% of students understood the concepts clearly (rated 5 and 4), whereas 7% found it moderately clear (rated 3), as shown in Figure 19. This translates to a mean rating of 4.59 and a standard deviation of 0.588, indicating that, despite the good grasping of the ML concepts, there were some minor variations in students' responses.

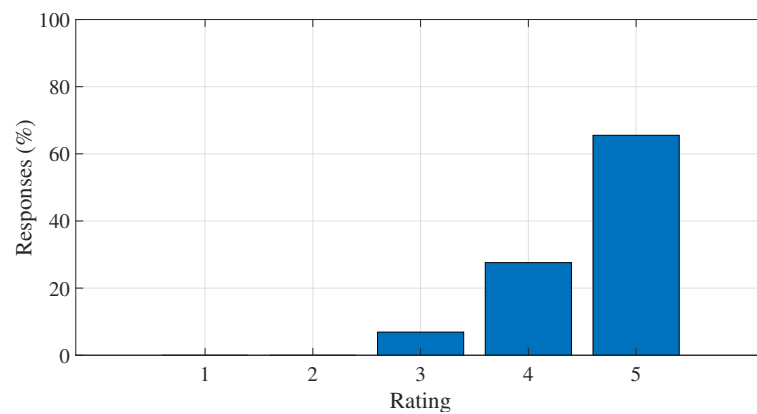


Figure 19. Q3 survey response.

The survey results indicate that the majority of students found it very easy or easy (rated 5 or 4) to design the use cases with the tool, as presented in Figure 20. Only very few of them found it neutral. This gives a mean of 4.76 and a standard deviation of 0.489, suggesting that designing use cases with the tool is manageable and easy.

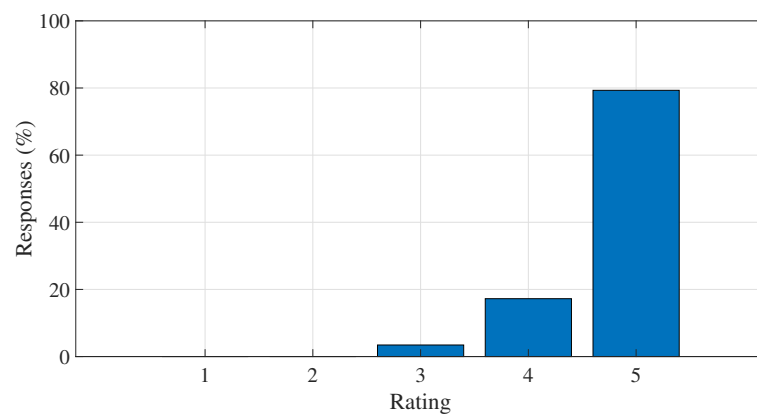


Figure 20. Q4 survey response.

Figure 21 provides insights into the tool's impact on participants' practical skills development for real-world applications. Approximately 62% expressed very high levels of satisfaction, with 34.5% rated 4, and only very few students rated 3. The statistical measures also indicate significant strides toward practical proficiency, with a mean of 4.59 and a standard deviation of 0.526.

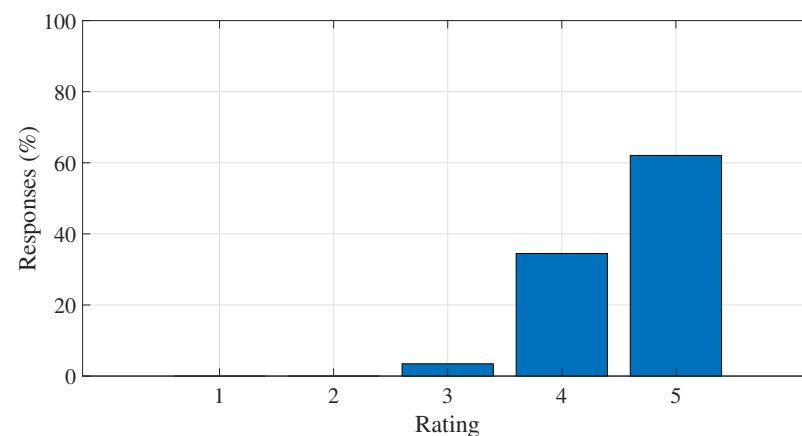


Figure 21. Q5 survey response.

A key aspect of the survey was to ask the students about the tool's user interface. All the students found it either extremely friendly or very friendly (rated 5 and 4), resulting in a mean of 4.62 and a standard deviation of 0.492. The survey results are shown in Figure 22.

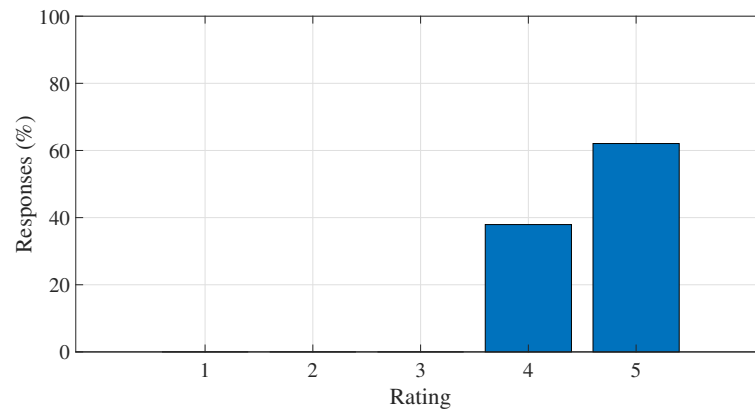


Figure 22. Q6 survey response.

5.2. Discussion

The survey findings underscore the positive impact of utilizing the IoT-Edu-ML-Stream platform to instruct electrical engineering students on the fundamentals of integrating IoT, big data, and ML.

The tool aids in grasping IoT concepts, as indicated by the feedback about the knowledge gained, as reported in Q2 results with an average rating of (4.66 ± 0.155) . Here, we used a 95% confidence interval to provide a meaningful analysis of the survey results of each question. This result indicates that the proposed tool enriched the students' knowledge of IoT concepts. This is despite the 62% expressed that they have good knowledge of IoT and fundamentals of ML, as indicated by an average rating of (3.62 ± 0.45) for Q1 results. This enhancement likely stems from the tool's amalgamation of hardware devices and IoT network configurations within its user interface.

However, there are areas for improvement highlighted by the responses to Q3, with an average rating of (4.59 ± 0.214) . The current version of the tool's limited support for ML algorithms may impede its effectiveness in comprehensively teaching ML principles. Additionally, the necessity for more time for participants to explore various use cases and scenarios suggests that further refinement and expansion of the tool's capabilities may be warranted to optimize its effectiveness.

Q4 serves the purpose of evaluating the tool user interface and the overall experience of designing a complete use case that requires IoT communication with big data analytics. The response to Q4 is reflected in an average rating of (4.76 ± 0.178) , which affirms the impact of the hands-on approach the tool follows in conveying the principles of designing a complete application use case from configuring the IoT-based network to the intelligent analytics of the collected data.

Furthermore, the majority of respondents highly rated the tool's efficacy in illustrating the practical interplay between IoT, big data, and ML, suggesting that it bolsters and enhances their existing knowledge in these domains, as indicated by an average rating of (4.59 ± 0.1915) for Q5 results. This observation emphasizes the significance of hands-on, experiential learning in reinforcing theoretical concepts, particularly in intricate and interdisciplinary fields like IoT and ML. Again, the need to spend more time with the tool in working out more use cases affected the survey response for this question, similar to Q3.

The friendliness of the tool user interface is measured by the response to Q6. The result (4.62 ± 0.179) reflects its intuitiveness and usability. This is crucial to reducing the students' dependency on the instructor, giving them self-confidence in learning the IoT, big data, and ML integration fundamentals.

In essence, the survey outcomes affirm that interactive and immersive learning environments, such as those facilitated by the proposed emulation platform, are indispensable for effectively conveying complex concepts in high-tech fields like IoT, big data, and ML. The survey results also imply that the proposed tool bridges the gap between academic theory and practical application by designing use cases addressing real-world scenarios. It furnishes students with invaluable skills and insights that qualify them for the practical job market. This is reflected in the high average rating of the survey questions, indicating the attainment of the proposed CLOs.

6. Conclusions and Future Directions

Although some AI-oriented courses are available in some curricula, they are for specific majors and mainly target AI techniques without addressing how these techniques can be integrated and applied in practice in essential disciplines such as electrical engineering. This makes electrical engineering students lack the knowledge and practice of IoT, machine learning (ML), and big data fundamentals and their relevance to the other courses they study in the mainstream curriculum.

Thus, this paper offers an emulation-based software platform for integrating IoT, ML, and big data into electrical engineering curricula. The proposed platform is a GUI-based tool called IoT-Edu-ML-Stream. By employing a friendly user interface, it allows students to design a wide range of use cases for data analytics by configuring an MQTT-based IoT network of sensor devices of various types, generating IoT data with a realistic pattern from these sensors, streaming it to a real-time big data platform, and finally performing intelligent ML-based analysis and decision-making.

Moreover, the paper introduces an example set of learning outcomes that can be addressed when using the tool in any course in an electrical engineering curriculum and their mapping to typical electrical engineering program outcomes. A survey was conducted among student participants to evaluate the tool. The results were analyzed, showing overall positive feedback regarding the tool usage and the knowledge gained.

In the future, the tool can be expanded to include other machine learning techniques (e.g., unsupervised learning), thereby increasing its applicability. Also, it can be modified to adopt other protocols for IoT communication. With these advancements, the IoT-Edu-ML-Stream tool should become an even more helpful teaching tool for IoT, big data, and ML education and experimentation.

Author Contributions: N.J. contributed to designing and developing the proposed tool. She also contributed to paper writing. N.Y. contributed to the development of the tool user interface. A.A. contributed to the architectural design of the proposed tool, the proposed lab tasks, and the course learning outcomes. He also participated in writing the paper. A.A.-D. contributed to fund acquisition and paper writing review. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the UAE University UPAR grant number 12N107.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, D.X.; Mendez, E.A. Artificial Intelligence in STEM Education: Interactive Hands-on Environment using Open Source Electronic Platforms. *Rev. Tecnol. En Marcha* **2023**, *36*, 45–52. [[CrossRef](#)]
2. Khan, M.A.; Khojah, M.; Vivek. Artificial intelligence and big data: The advent of new pedagogy in the adaptive e-learning system in the higher educational institutions of Saudi Arabia. *Educ. Res. Int.* **2022**, *2022*, 1263555. [[CrossRef](#)]
3. Liu, Y.; Chen, L.; Yao, Z. The application of artificial intelligence assistant to deep learning in teachers' teaching and students' learning processes. *Front. Psychol.* **2022**, *13*, 9175. [[CrossRef](#)] [[PubMed](#)]
4. Mannodi-Kanakithodi, A.; McDannald, A.; Sun, S.; Desai, S.; Brown, K.A.; Kusne, A.G. A framework for materials informatics education through workshops. *MRS Bull.* **2023**, *48*, 560–569. [[CrossRef](#)]

5. Kantawong, K.; Chaichumpa, S.; Pravesjit, S.; Yaibuates, M. A Lightweight Framework for Retrieve IP Device Status Based on MQTT Protocol. In Proceedings of the 2022 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), Chiang Rai, Thailand, 26–28 January 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 46–49. [[CrossRef](#)]
6. Foundation, A.S. Apache Kafka. Available online: <https://kafka.apache.org/> (accessed on 4 December 2023).
7. Sahu, C.; Ayotte, B.; Banavar, M.K. Integrating machine learning concepts into undergraduate classes. In Proceedings of the 2021 IEEE Frontiers in Education Conference (FIE), Lincoln, NE, USA, 13–16 October 2021; pp. 1–5. [[CrossRef](#)]
8. Filsinger, M. Designing an Artificial Intelligence Course for Electrical and Computer Engineering Technology Students. In Proceedings of the 2006 Annual Conference and Exposition, Saint Louis, MO, USA, 4–7 June 2006; pp. 11–417. [[CrossRef](#)]
9. Montuori, L.; Alcazar-Ortega, M.; Vargas-Salgado, C.; Alfonso-Solar, D. Learning Analytics as Data-driven Decision Making in Higher Education: A Case Study. In Proceedings of the INNODOCT 2022. International Conference on Innovation, Documentation and Education, Valencia, Spain, 2–7 November 2022; pp. 83–89. [[CrossRef](#)]
10. Starcic, A.I. Human learning and learning analytics in the age of artificial intelligence. *Br. J. Educ. Technol.* **2019**, *50*, 2974–2976. [[CrossRef](#)]
11. Tan, J. Information analysis of advanced mathematics education-adaptive algorithm based on big data. *Math. Probl. Eng.* **2022**, *2022*, 7796681. [[CrossRef](#)]
12. Barenkamp, M.; Rebstadt, J.; Thomas, O. Applications of AI in classical software engineering. *AI Perspect.* **2020**, *2*, 1. [[CrossRef](#)]
13. Wei, N. Decreasing land use and increasing information infrastructure: Big data analytics driven integrated online learning framework in rural education. *Front. Environ. Sci.* **2022**, *10*, 1025646. [[CrossRef](#)]
14. Xu, L.; Zhang, J.; Ding, Y.; Sun, G.; Zhang, W.; Philbin, S.P.; Guo, B.H. Assessing the impact of digital education and the role of the big data analytics course to enhance the skills and employability of engineering students. *Front. Psychol.* **2022**, *13*, 974574. [[CrossRef](#)]
15. Tsai, S.C.; Chen, C.H.; Shiao, Y.T.; Ciou, J.S.; Wu, T.N. Precision education with statistical learning and deep learning: A case study in Taiwan. *Int. J. Educ. Technol. High. Educ.* **2020**, *17*, 12. [[CrossRef](#)]
16. Gil, Y. Teaching big data analytics skills with intelligent workflow systems. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, Arizona, 12–17 February 2016; Volume 30. [[CrossRef](#)]
17. Panagiotakis, S.; Karampidis, K.; Garefalakis, M.; Tsironi-Lamari, A.; Rallis, I.; Kamarianakis, Z.; Papadourakis, G. Remote Arduino Labs for Teaching Microcontrollers and Internet of Things Programming. In Proceedings of the 31st Annual Conference of the European Association for Education in Electrical and Information Engineering (EAEIE), Coimbra, Portugal, 29 June–1 July 2022; pp. 1–6. [[CrossRef](#)]
18. Hercog, D.; Lerher, T.; Truntič, M.; Težak, O. Design and Implementation of ESP32-Based IoT Devices. *Sensors* **2023**, *23*, 6739. [[CrossRef](#)]
19. Hassan Al-Taai, S.H.; Abbas Kanber, N.H.; Mohammed al Dulaimi, W.A. The Importance of Using the Internet of Things in Education. *Int. J. Emerg. Technol. Learn. (Ijet)* **2023**, *18*, 19–39. [[CrossRef](#)]
20. Ghazanfar, S.; Hussain, F.; Rehman, A.U.; Fayyaz, U.U.; Shahzad, F.; Shah, G.A. IoT-Flock: An Open-source Framework for IoT Traffic Generation. In Proceedings of the 2020 International Conference on Emerging Trends in Smart Technologies (ICETST), Karachi, Pakistan, 26–27 March 2020; pp. 1–6. [[CrossRef](#)]
21. Alammary, S.; Alhazmi, S.; Almasri, M.; Gillani, S. Blockchain-based Applications in Education: A Systematic Review. *Appl. Sci.* **2019**, *9*, 2400. [[CrossRef](#)]
22. Karale, A.; Khanuja, H. Implementation of Blockchain Technology in Education System. *Int. J. Recent Technol. Eng.* **2019**, *8*, 2823–2828. [[CrossRef](#)]
23. Shmatko, O.; Borova, T.; Yevseiev, S.; Milov, O. Tokenization of Educational Assets Based on Blockchain Technologies. *Sci. Pedagog. Educ.* **2021**, *3*, 4–10. [[CrossRef](#)]
24. Han, L. Exploring the Mode of Inter-University General Education Consortium Based on Blockchain Technology. *J. Educ. Educ. Res.* **2023**, *6*, 113–115. [[CrossRef](#)]
25. Uspabayeva, A.; Sattarova, A.; Mirza, N.; Kubeeva, M.; Abdualievich, Z.K.; Rysbayeva, G. Evaluation of High School Students' New Trends in Education: Internet of Things. *Int. J. Emerg. Technol. Learn. (Ijet)* **2022**, *17*, 159–175. [[CrossRef](#)]
26. Sang, Y.; Wang, L. Physical Fitness Data Monitoring of College Students Based on the Internet of Things and Blockchain. *Front. Public Health* **2022**, *10*, 940951. [[CrossRef](#)]
27. Yujie, H.; Qiuxia, H. Innovative Mode of Logistics Management of “Internet of Things + Blockchain”-Integrated E-Commerce Platform. *Comput. Intell. Neurosci.* **2022**, *2022*, 7766228. [[CrossRef](#)]
28. Shahin, Y.M. Technological Acceptance of the Internet of Things (IOT) in Egyptian Schools. *Int. J. Instr. Technol. Educ. Stud.* **2020**, *1*, 6–10. [[CrossRef](#)]
29. Jeong, Y.S. Secure IIoT Information Reinforcement Model Based on IIoT Information Platform Using Blockchain. *Sensors* **2022**, *22*, 4645. [[CrossRef](#)] [[PubMed](#)]
30. Kyun, S.; JaeKyung, Y.; Jang, J.Y. A Decentralized Approach to Education Powered by Blockchain Technology. *Asia-Pac. J. Converg. Res. Interchange* **2021**, *7*, 131–141. [[CrossRef](#)]
31. Wang, Z.; Wu, W.; Zeng, C.; Yao, J.L.; Yang, Y.; Xu, H. Graph Neural Networks Enhanced Smart Contract Vulnerability Detection of Educational Blockchain. *arXiv* **2023**, arXiv:2303.04477.

32. Miorandi, D.; Sicari, S.; De Pellegrini, F.; Chlamtac, I. Internet of things: Vision, applications and research challenges. *Ad Hoc Netw.* **2012**, *10*, 1497–1516. [[CrossRef](#)]
33. KimiNewt. PyShark. Available online: <https://pypi.org/project/pyshark/> (accessed on 4 December 2023).
34. Sartori, I.; Walnum, H.T.; Skeie, K.S.; Georges, L.; Knudsen, M.D.; Bacher, P.; Candanedo, J.; Sigounis, A.M.; Prakash, A.K.; Pritoni, M.; et al. Sub-hourly measurement datasets from 6 real buildings: Energy use and indoor climate. *Data Brief* **2023**, *48*, 13. [[CrossRef](#)]
35. ABET. Criteria for Accrediting Engineering Programs, 2019–2020. Available online: <https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2019-2020/> (accessed on 4 December 2023).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.