Boulez: A Chatbot-based Federated Learning System for Distance Learning

Stefano D'Urso
Faculty of Economics,
Universitas Mercatorum,
Piazza Mattei, 10
Rome, Italy
stefano.durso@studenti.unimercatorum.it

Filippo Sciarrone
Faculty of Economics,
Universitas Mercatorum,
Piazza Mattei, 10
Rome, Italy
filippo.sciarrone@unimercatorum.it

Marco Temperini
Department of Computer, Control, and Management Engineering,
Sapienza, University of Rome,
Via Ariosto 25,
Rome, Italy
marte@diag.uniroma1.it

***Abstract*—In recent years, also due to the covid-19 pandemic, the possibilities for distance learning have increased considerably, through web-based learning platforms, available on the Internet without space and time limits. As a result, the offer of courses and the number of enrolled students has grown exponentially. In order to be able to guarantee students a better learning support service, one of the proposals regards the intelligent Chatbots. These well known interactive applications are based mainly on machine or deep learning and in this paper we present Boulez, a system allowing the orchestration of a community of individual chatbots, each one with its algorithm and its private training dataset. We apply a technique called Federated Learning, where several individual chatbots, collaborate. In particular, here the approach is "centralized", meaning that a main system orchestrates the collaboration of the federated systems. By addressing the communication inefficiencies and privacy issues of conventional federated learning, Boulez offers a more efficient and effective approach to chatbot interaction, ultimately leading to improved user experience. The paper presents the Boulez system, its operation principle, methods used, and potential benefits, along with a use case of its application.**

***Index Terms*—Intelligent Chatbot, Federate Learning, Machine Learning.**

I. Introduction

In recent years there has been a sharp increase in distance learning. This process, which began mainly because of the COVID-19 pandemic, has brought major changes in learning and teaching modalities around the world. For istance, Massive Open Online Courses (MOOC)s, represent the concrete effect of these changes, with courses having huge enrollment numbers and continuously growing, such as *Coursera*[1], *Udemy*[2] and *Udacity*[3], just to name a few of the most popular ones, with tens or hundreds of thousands of members. However, many public and private institutions, like schools and universities have adopted distance learning as the unique teaching and learning methodology. Consequently, more and more courses delivered in a remote modality have been attended by a huge number of people, producing an increasing number of MOOCs and imposing new challenges and problems for all the stakeholders [9]. One of the problems in such courses is that concerning the support given to students: due to the huge number of learners, teachers cannot directly support each of them personally, and, consequently, the *one-size-fits-all* didactic approach is used [13]. For this reason, and also thanks to the development of Deep Learening (DL), chatbots are being proposed, conceived and designed to support students in their learning process [10], [11]. Most of the chatbots are built by using deep learning models to train them. ChatGPT is the most famous example of this approach [12]. *Federated Learning* (FL), is a popular technique that enables collaborative machine learning in distributed systems, where data is stored locally on multiple devices and is not directly accessible to a central server [5], [16] . This approach has gained significant attention in recent years, in both industry and academia, as it enables organizations to train local machine learning models safeguarding private and privacy data, without the need to share it with others [1], [5], [7], [15]. Moreover, there has been growing interest in the development of new strategies for improving FL systems. One of such strategies is based on Artificial Intelligence (AI) techniques, which can help to optimize the learning process by leveraging the collective intelligence of the distributed nodes end data [7]. In this paper, we introduce a novel FL-based system, called *Boulez*, which aims to orchestrate a community of chatbots, each belonging to a single node. A single node could be a class or a set comprising even a large number of courses. The Boulez system addresses the problem of how to let different

[1]https://www.coursera.org/
[2]https://www.udemy.com/
[3]www.udacity.com

chatbots to communicate each other to better improve students' learning within a specific class, by means of an enhanced service. By federating chatbots, Boulez enables organizations to collaboratively train machine learning models using their local data and expertise, while preserving their privacy and security. This approach has potential applications in the educational field and other domains where personalized and scalable machine learning is needed. So, through the Boulez system, organizations can request and offer contributions to a shared machine learning model based on chatbots, while an orchestrator manages the communication between them. The system is currently at its early stage of development and here we present its architecture, with a useful use case. Section II briefly introduces some important works both on intelligent chatbots and FL, giving some background as well. Section III shows the architecture of the Boulez system. In section IV a use case of the use of the system is shown, while in section V, conclusions are drawn.

II. Related Work and Background

Few works have been proposed concerning the use of FL architectures in the educational field or for chatbots orchestration. So, in this section, we give some background on both FL and intelligent chatbots as well.

In [2], the authors present a proof-of-concept privacy-preserving chatbot that leverages large-scale customer support data. They use FL as a method for distributed and collaborative machine learning to train large models for chatbots and conversational agents. In particular, a FL model, called *FedBot* is proposed to respect data privacy regulations. Differenty, our system aims to work in educational contexts only, in order to improve students' learning in all the FL nodes. To this aim the Boulez's FL architecture is different from the classic FL ones.

In [14], the authors propose a FL-based education data analysis framework called *FEEDAN*, via which education data analysis federations can be formed by a number of institutions. The framework is used to analyze two real education data sets via two different federated learning paradigms. Differently our system aims at improving students' learning by a community of chatbots orchestrated as in a FL architecture.

A. Federate Learning

FL is a distributed machine learning concept introduced by Google in 2016 [6]. It involves multiple clients, each of them with its own data for a particular task, without direct access to other clients' data. The objective of FL is to learn a predictive central model that minimizes the error on an objective function in a distributed way [15]. FL has gained substantial interest in the machine learning community, with different frameworks implementing the main concept and applications becoming more frequent. The advantages of FL over classical local machine learning nodes, include the ability to employ the power of distributed client machines, keep user data private, and use information that would be otherwise inaccessible and spread over different clients. FL aligns with recent trends on machine learning on large community data and increasing constraints due to privacy regulations and trustworthy AI. In the FL context, it is generally assumed that each user provides a sufficient set of labeled data for a model to learn a specified task, and that the analyzed data is given in common feature spaces, although potentially split across various clients. When these assumptions do not hold, Federated Transfer Learning (FTL) methods have to be applied [8]. FTL helps improve the model of a target user, by using data or model information from one or more source users.

B. Intelligent Chatbots

Conversational agents or chatbots, powered by AI have become increasingly popular in industries, including e-commerce, online banking, healthcare and education. However, the use of chatbots in education remains limited [10]. Chatbots have the potential to provide personalized service and support to students and teachers, overcoming the one-size-fits-all approach. There are examples of chatbot prototypes being developed by the *Warwick Manufacturing Group*, at the University of Warwick to support educational activities, such as the delivery of a taught Master's course simulation game, training for a new educational application, and processing helpdesk requests [16]. Recent developments in Generative Pre-trained Transformers (GPT) models, such as ChatGPT, have enhanced the capabilities of chatbots in education [4]. GPT models have significantly improved natural language processing, enabling chatbots to have more human-like conversations with students and teachers. The language generation capabilities of GPT models also make it possible for chatbots to generate personalized responses and provide tailored support to students [3]. Additionally, GPT models can analyze large amounts of data and identify patterns and trends, enabling chatbots to provide more accurate and informed recommendations to students and teachers. These advancements in GPT models have the potential to revolutionize the use of chatbots in education, making them even more effective in supporting teaching and learning activities. Each node represents an autonomous software system, consisting of chatbot(s) capable of handling incoming requests within the local node. The main characteristic of our system is the use of a FL approach to orchestrate chatbots in the educational field. This goal allows us to specialize the system through the use of modules dedicated to improving student learning belonging to individual nodes orchestrated by the Boulez system.

III. The Boulez System

With the recent advancements in FL and chatbots, there has been a growing interest in exploring their potential applications in various fields, including education: the combination of these two technologies has shown promising results towards the improvement of personalized learning experiences for students, while preserving the privacy of their data [2], [14]. However, to fully exploit the benefits of these technologies, new architectures that can efficiently manage and coordinate chatbots across different nodes are required.

In this section we describe the Boulez system, whose architecture is shown in fig. 1. The name of the project is inspired by the famous French composer and conductor Pierre Boulez, and aims to convey the idea of a software system that orchestrates and directs the completion of tasks and requests in educational context which use chatbots.

So, the individual nodes, in the Boulez system, are chatbots which are connected to a central node (the proper Boulez module), where an *Orchestrator* and other modules support the nodes collaboration, i.e., the chatbots community.

As shown in fig. 1, the Boulez system is composed of an *Orchestrator* , a *Web interface*, a *Trust* module and an *Internal Storage* component. Each module acts as follows:

- The orchestrator's main task is to manage incoming requests from those connected nodes that require a completion of a prompt, for which they do not have an appropriate or entirely relevant completion of their own, and to select the best completion provided by the nodes to subsequently provide it to the requester, i.e. the learner;
- The web interface is responsible for receiving requests from the connected nodes, allowing for seamless communication between the Boulez orchestrator and each individual chatbot system;
- The trust module assigns a level of trust to each connected node based on feedbacks produced by individual nodes towards others and based on periodic requests that Boulez makes to individual nodes. For a better understanding we can imagine this operation as the *ping* command in a network;
- The internal storage component is a critical component that ensures the persistence of information related to the connected nodes, including their identity, registry, and other relevant metadata: this component plays a vital role in maintaining the integrity and reliability of the chatbots by enabling accurate tracking and management of node-specific information.

By Boulez, users can rely on an efficient and effective management of chatbots' interactions, resulting in a smoother and more satisfying user experience. Each node represents an autonomous software system, consisting of chatbot(s) capable of handling incoming requests within the local node. These chatbots will be trained through proprietary systems or integrated with external APIs (such as NLP services from OpenAI, Google, or Amazon), or other architectures chosen by the node's owner, allowing them to understand and respond to learners' queries. However, it may happen that not all the students' requests posted to their chatbot, can be answered adequately by the chatbot itself: this is just the case where Boulez supports the chatbot to replay to the student. In fact, when a node needs assistance, it can submit a request for completion to Boulez by contacting the Web API exposed by Boulez. Upon receiving the request, Boulez will contact all the connected nodes in order to select the *best* response, transmitting it to the requester.

For a node to be managed by Boulez, it must be able to perform the *transmit and receive* operations. This means that a node can *receive* completions (of missing answers) through the web API offered by Boulez, and should be able to *transmit* completions, when requested by Boulez, through a suitable communication protocol managed by the Boulez Web API. In this way, each node could contribute to the overall efficiency and effectiveness of the chatbot's community, ensuring that the best possible completions, i.e., answers, are selected and provided to the learner. Boulez will manage the process of receiving completion requests, executing a broadcast call to all the connected nodes, waiting for their responses (according to a predetermined timeout), and selecting the best response basing on its *accuracy* parameter that each node must provide along with the completion offered. In this way, Boulez ensures that the most appropriate completion is selected and provided to the requester, based on the input from all connected nodes.

In addition to its core functionalities, Boulez also employs an additional module called the *Trust Module* (see fig. 1), which assigns a level of trust to each connected node based on the feedbacks produced by individual nodes towards others (following responses received through the orchestrator) and based on periodic requests that Boulez transmit to individual nodes. These requests' outcomes are evaluated through an internal procedure, allowing Boulez to continuously monitor the performance of each node and adjust the level of trust accordingly.

By the Trust Module, Boulez ensures that only the most reliable and effective nodes are utilized in completing user requests, further enhancing the overall performance and reliability of the overall chatbot network. To select the best response, among those provided by the nodes, Boulez weights both the accuracy communicated by the individual node about its completion, and the trust level assigned, and maintained for that node by the central system. Consequently, basing on the Trust module, Boulez can more accurately assess the reliability and effectiveness of each node's responses, leading to better overall performance and a more seamless user experience.

A. The Software Components

As mentioned above, the Boulez's main task is to manage incoming requests from connected nodes that require a completion of a prompt, for which they do not have an appropriate or entirely relevant completion on their own, and to *select* the best completion provided by the nodes to subsequently provide it to the requester.

In order for a node to integrate with Boulez, it must expose a web API that allows Boulez to verify if there is a completion for the requested prompt and what the accuracy of the response is. Fig. 1 shows the generic node software components. The *GetCompletion* API function is specifically designed to fulfill this requirement, providing a secure and efficient method for Boulez to receive and process completion requests from connected nodes. The GetCompletion API accepts *POST* requests in the form of a *JSON* object containing the following three parameters:

- *Prompt*: a text string representing the prompt for completion. This parameter is mandatory and cannot be empty;

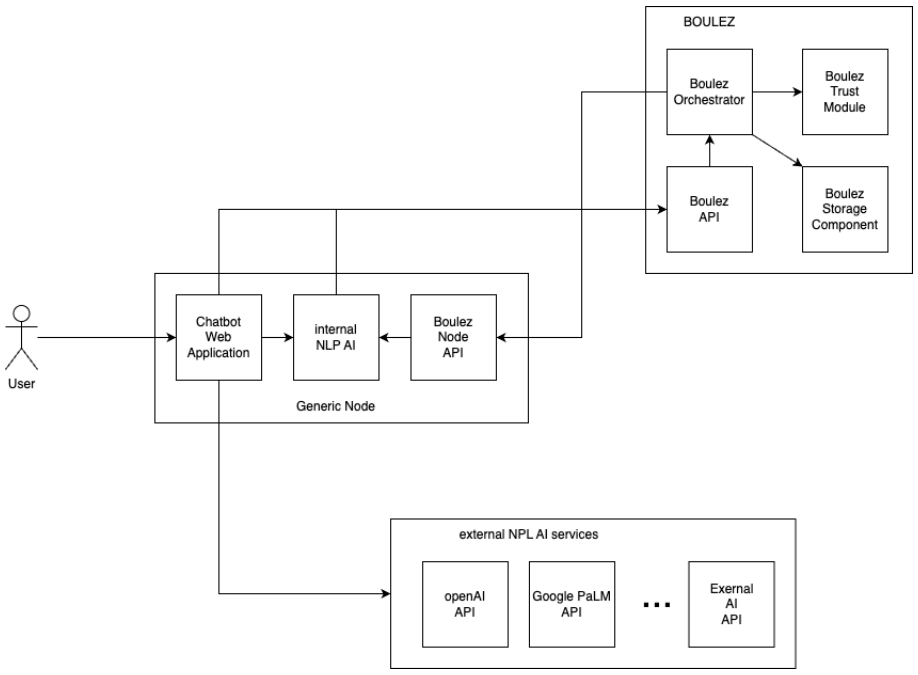

Fig. 1. The architecture of the Boulez system.

- *Request Id*: a unique identifier for the request, generated by an external partner software. This parameter is mandatory and must be unique for each request;
- *Timestamp*: a string representing the date and time of the request. This parameter is mandatory.

Consequently, the GetCompletion API returns a JSON response containing the following four parameters:

- *Completion*: a text string representing the completion generated by the system;
- *Response Id*: a unique identifier for the response, generated by the system. This parameter is the same as the answer id of the request passed as input;
- *Timestamp*: a string representing the date and time of the response.
- *Accuracy*: a numerical value representing the accuracy of the completion, expressed as a percentage.

So, the Boulez GetCompletion API function, allows a node to request a completion from other connected nodes through the Boulez system. Boulez provides and additional Feedback API that could be used by each node in order to provide feedback to external completions provided by other nodes in the federated system. The *Feedback* API accepts POST requests in which a JSON object containing two parameters is passed:

- *Completion Id*: a unique identifier of the completion, generated by the system. This parameter is required and must be valid;
- *Rating*: an integer value representing the rating of the completion, where -1 indicates an incorrect or inadequate response, 0 indicates a neutral or partially adequate response, and +1 indicates a correct or highly adequate response. This parameter is required;

The API returns a JSON response containing one parameter:

- *Item Status*: it is an integer value indicating the status of the operation, with 0 representing success and -1 representing an error.

In addition to the real-time feedback mechanism provided by the Feedback API, Boulez also offers a periodic offline feedback mechanism that enables nodes to provide feedback at their own pace. In this process, Boulez sends a list of completed requests, along with the responses provided by each connected node, to the node via email or FTP at regular intervals. The node's users can then review this list and provide feedback on each response by clicking on a pre-set link in the email or document. The link leads to a simple form that allows the user to select the appropriate rating for each response (i.e., -1 for incorrect or inadequate, 0 for neutral or partially adequate, and +1 for correct or highly adequate). Once the user submits the form, the feedback is processed by Boulez and used to improve the quality of future responses.

IV. Use case: Educational Federated Learning

In this section we show a use of the Boulez architecture, by sketching the framework for an Educational application of FL (EdFL, Fig. 2).

In fig. 2 the elements of the EdFL framework are shown: the EdFL module, that manages the FL architecture, 2) a sample node, Chatbot_UNI_i, i.e. the i^{th} Higher Education (HE) node; and 3) several other nodes, of different types, such as other HE nodes, High School (HiS) nodes, and Elementary School (ELS) nodes. All nodes share the same structure and aims: a chatbot serves the node students about their questions, based on the *Local Learning Material* (LM) provided by the

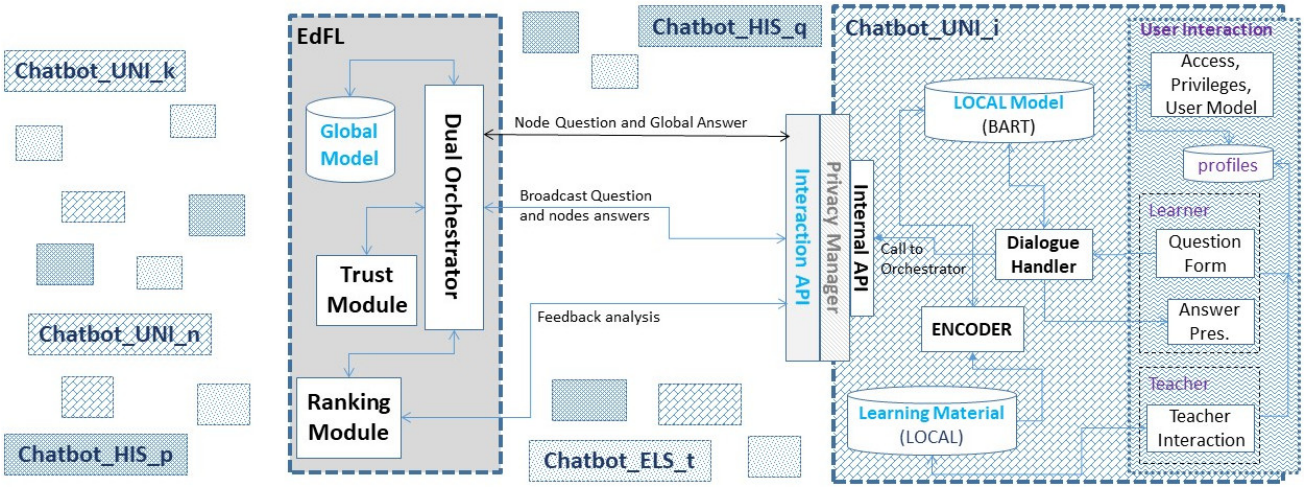

Fig. 2. Architecture of an Educational Federated Learning application based on Boulez.

teachers of that node. In each node such material concurs to the formation of a *Local Model*, managed according to Boulez specifications (for example using BART as the deep learning model). Based on such modeling of the Local Learning Material, the questions coming from the students are answered to. All nodes shown in figure, as well, have the same connections and relations with the EdFL module, as the sample node, which is the only one the figure shows in detail. Each node has its own interface, which differently may give the students the possibility to formulate questions and receive the answers. In figure that is comprised in the *User Management* module of the node. The *Dialog Handler* module handles the whole dialogue with a student: when it receives a question from the interface, it interacts with the Local Model, contextualizes to the current dialogue, develops comprehension of the question, and build an answer. It also computes a *Local Accuracy* of the answer (with respect to the Local Model): If the accuracy is above a pre-defined *Local Accuracy Threshold*, the answer is then passed to the *Answer Presentation module* in the user interface. The interface also allows the student to express satisfaction or dissatisfaction with the answer. When the answer to a question is not found in the Local Model, or it is not found with sufficient Local Accuracy, or the student's feedback is of dissatisfaction, the question is re-routed to the EdFL Module, and its *Dual Orchestrator*. The EdFL module contains the *Dual Orchestrator*, and maintains the nodes profiles, by managing the *Global Model*, obtained from the nodes' Local Models, the *Trust Module*, and in addition managing a *Ranking Module*. The Orchestrator is named "Dual" as it manages, in effect, both the orchestration of federated nodes and the interaction among the inner modules of the EdFL module. The Ranking Module collects, by interacting with each individual node, information about 1) the students' feedback on the answers provided by the node, 2) the Local Accuracy of the node answers, and 3) a *Global Accuracy* computed on the answers given by the node, based on the Global Model. The mentioned information are the ground on which a ranking of the nodes is maintained, to be used by the Dual Orchestrator. When the Dual Orchestrator receives a question from a node (Originator node), it broadcasts a request to all the relevant nodes, that is all those with sufficient trust and rank among the nodes of the same type of the Originator (e.g. all HIS, if the node sending the question is HIS), and collect the answers. Then the answers are ranked by Global Accuracy and, depending on the request of the Originator, the best or some of the best, are sent back to the Originator. In this section we gave a basic educational use case for the previously described Boulez system. The EdFL architecture presents no alternatives for the students, beside the direct answer provided by their node, or the global answer(s) provided by the whole nodes federation. The Boulez system supports further interactions, with external NLPAI services, to keep looking for an answer: that can be a powerful extension of EdFL. Such extension, though, should allow the individual node to have its say in the way it would work. In particular, it is expectable that the individual educational institution (federated node) would like to 1) allow for students to search through learning landscapes beyond the one provided by their institution, while 2) managing a degree of consistency of the student's learning activity with the learning objectives of the study program. So it is conceivable that the extension with additional NLPAI systems should be regulated at node level. All this is seen as future work in the enhancement of EdFL.

V. Conclusions and Future Work

We have presented Boulez, a FLg system where nodes are chatbots, that are made collaborate, where needed, to complete an answer that could not be provided to satisfaction by an individual chatbot in the network to an individual learner belonging to a particular class.

By using FL and a centralized orchestration approach, Boulez enables chatbots to share knowledge with each other and collectively improve their performance over time. Moreover, Boulez ensures privacy by allowing each node to keep its

data private while still benefiting from the insights shared by other nodes. The chatbot coordination pursued through Boulez has the potential to allow for more efficient and personalized interactions with users.

While Boulez is designed to support every kind of FL networks, educational applications were of interest in the present paper, and we described a concept design of *Educational Federated Learning - EdFL*, aimed to manage networks where several educational institutions, of varied levels, can collaborate and support each other, by sharing the products (answer completions) of their local chatbot-based services. The system is at its early stage of development and for future work we plan to finish its actual implementation in order to run a field evaluation as soon as possible.

REFERENCES

[1] Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. p. 308–318. CCS '16, Association for Computing Machinery, New York, NY, USA (2016)

[2] Ait-Mlouk, A., Alawadi, S., Toor, S., Hellander, A.: Fedbot: Enhancing privacy in chatbots with federated learning (2023)

[3] au2, M.B.I., Katz, D.M.: Gpt takes the bar exam (2022)

[4] Bavarian, M., Jun, H., Tezak, N., Schulman, J., McLeavey, C., Tworek, J., Chen, M.: Efficient training of language models to fill in the middle (2022)

[5] Heusinger, M., Raab, C., Rossi, F., Schleif, F.M.: Federated learning - methods, applications and beyond. In: ESANN 2021 proceedings. Ciaco - i6doc.com (2021)

[6] Konecný, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: Federated learning: Strategies for improving communication efficiency. ArXiv **abs/1610.05492** (2016)

[7] Li, L., Fan, Y., Tse, M., Lin, K.Y.: A review of applications in federated learning. Computers & Industrial Engineering **149**, 106854 (2020)

[8] Liu, Y., Kang, Y., Xing, C., Chen, T., Yang, Q.: A secure federated transfer learning framework. IEEE Intelligent Systems **35**(4), 70–82 (2020). https://doi.org/10.1109/MIS.2020.2988525

[9] Norfarahi, Z., Mohd, I.H., Nur, H.B.: Challenges to teaching and learning using mooc. Creative Education **11**(3) (2020)

[10] Okonkwo, C.W., Ade-Ibijola, A.: Chatbots applications in education: A systematic review. Computers and Education: Artificial Intelligence **2**, 100033 (2021)

[11] Okonkwo, C.W., Ade-Ibijola, A.: Chatbots applications in education: A systematic review. Computers and Education: Artificial Intelligence **2**, 100033 (2021)

[12] OpenAI: Gpt-4 technical report (2023)

[13] Shrivastava, A., Shrivastava, A., Bhatt, V., Sinha, B.: Moocs - one size does not fit all. In: 2022 Interdisciplinary Research in Technology and Management (IRTM). pp. 1–5 (2022). https://doi.org/10.1109/IRTM54583.2022.9791519

[14] Song Guo, Deze Zeng, S.D.: Pedagogical data analysis via federated learning toward education 4.0. American Journal of Education and Information Technology **4**, 56–65 (2020)

[15] Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: Concept and applications **10**(2) (jan 2019)

[16] Yang, S., Evans, C.: Opportunities and challenges in using ai chatbots in higher education. In: Proceedings of the 2019 3rd International Conference on Education and E-Learning. p. 79–83. ICEEL '19, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3371647.3371659, https://doi.org/10.1145/3371647.3371659