Facilitating the Learning Engineering Process for Educational Conversational Modules Using Transformer-Based Language Models

Behzad Mirzababaei and Viktoria Pammer-Schindler

***Abstract*—In this article, we investigate a systematic workflow that supports the learning engineering process of formulating the starting question for a conversational module based on existing learning materials, specifying the input that transformer-based language models need to function as classifiers, and specifying the adaptive dialogue structure, i.e., the turns the classifiers can choose between. Our primary purpose is to evaluate the effectiveness of conversational modules if a learning engineer follows our workflow. Notably, our workflow is technically lightweight, in the sense that no further training of the models is expected. To evaluate the workflow, we created three different conversational modules. For each, we assessed classifier quality and how coherent the follow-up question asked by the agent was based on the classification results of the user response. The classifiers reached F1-macro scores between 0.66 and 0.86, and the percentage of coherent follow-up questions asked by the agent was between 79% and 84%. These results highlight, first, the potential of transformer-based models to support learning engineers in developing dedicated conversational agents. Second, it highlights the necessity to consider the quality of the adaptation mechanism together with the adaptive dialogue. As such models continue to be improved, their benefits for learning engineering will rise. Future work would be valuable to investigate the usability of this workflow by learning engineers with different backgrounds and prior knowledge on the technical and pedagogical aspects of learning engineering.**

***Index Terms*—Adaptive learning technology, argument mining, educational conversational agent, learning engineering, Toulmin's model of argument, transformer-based language models.**

I. Introduction

AN EDUCATIONAL conversational agent needs content and an adaptation mechanism to be effective in educational scenarios. The educational content should be carefully structured and aligned with the adaptation mechanisms. From a technical perspective, the adaptation mechanisms behind an educational agent play a crucial role in its effectiveness. These mechanisms enable the agent to analyze and understand the user's inputs and provide personalized educational content.

In a typical educational scenario, it is essential for learners to not only understand a concept but also be able to apply it and argue about it effectively. In the field of education, argumentation tasks, such as posing argumentative questions, play a crucial role in deepening students' knowledge in specific subject domains [1]. Students not only need to develop valid arguments but also engage in scientific reasoning through argumentation [2]. In general, argumentation serves as a heuristic method for developing a deeper understanding of scientific concepts [1]. Effective questions should elicit argumentation, thereby stimulating creativity and critical thinking, and boosting students' confidence [3].

In this work, we are interested in conversational modules that can do the following:

1) provide a concept or a definition from the student's learning domain;
2) ask the learner to apply this definition to an example;
3) give adaptive feedback to the learners on their reasoning.

Such questions are central to students' cognitive development, and research evidence suggests that students' levels of achievement can be increased by regular access to higher order thinking (e.g., [4]).

For instance, in the learning domain of astronomy, given the definitions of a planet and a star, learners would be asked to apply the definitions to a specific example, such as Jupiter, and reason about it. The question asked by the agent would be "*Based on the definitions, is Jupiter a planet or a star? Explain why?*" A complete answer from a learner needs to include a claim ("*Yes, Jupiter is a planet*"). Learners also need to make a connection between Jupiter and the definitions, e.g., by citing the given definition of a planet and providing evidence related to Jupiter showing that it fulfills the definition. Fig. 1 shows another example, based on real user data, in the domain of European General Data Protection Regulation (GDPR). In Fig. 1, note that the last question of the agent responds adaptively to the user statement.

The adaptation mechanism selects follow-up questions if the learner does not reason sufficiently. In this work, we base our understanding of what constitutes a sufficiently reasoned answer on Toulmin's model of argument [5]. Concretely, we expect there to be a claim, i.e., whether the example fulfils a given

Manuscript received 28 September 2023; revised 12 January 2024 and 13 February 2024; accepted 14 February 2024. Date of publication 20 February 2024; date of current version 1 March 2024. This work was supported in part by the COMET—Competence Centers for Excellent Technologies Program, through the Austrian Federal Ministry (BMK and BMDW), in part by DDAI COMET Module, in part by Austrian Research Promotion Agency (FFG), in part by the province of Styria (SFG), and in part by partners from industry and academia. *(Corresponding author: Behzad Mirzababaei.)*

Behzad Mirzababaei and Viktoria Pammer-Schindler are with Know-Center GmbH, 8010 Graz, Austria, also with the Graz University of Technology, 8010 Graz, Austria (e-mail: bmirzababaei@student.tugraz.at).

This article has supplementary downloadable material available at https://doi.org/10.1109/TLT.2024.3367738, provided by the authors.

Digital Object Identifier 10.1109/TLT.2024.3367738

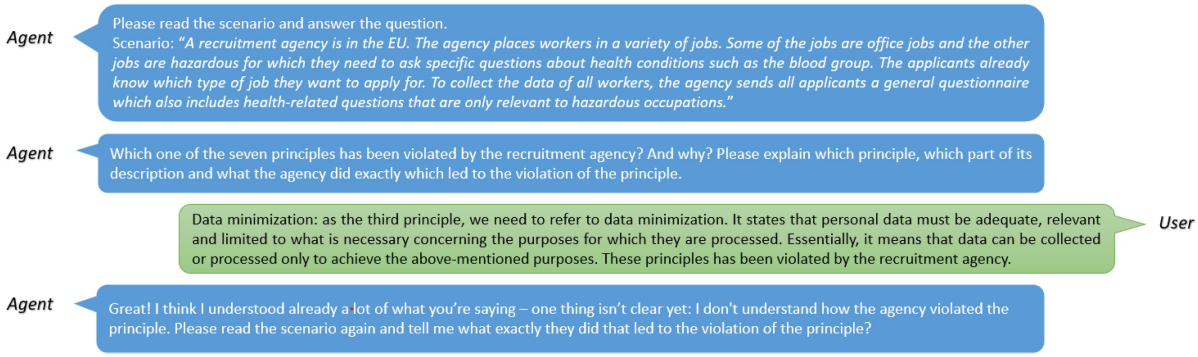

Fig. 1. Conversational module about the topic of the GDPR created with the workflow. The user's response is a real answer to the question.

definition, a warrant, i.e., an explicit mention of relevant parts of the definition, and evidence, i.e., an explicit relationship between the example and the definition. This corresponds to the core elements in Toulmin's model of arguments (ibid). Prior work shows this to be a suitable structure for argumentative answers or essays (e.g., [6] and [7]), even if other models of argument also exist (e.g., [8]).

We call such a system "a conversational module" to clarify that we think of it not necessarily as a stand-alone system, but more probably as a partial system that needs to be part of a wider educational intervention. The conversational module could be a stand-alone system but used within a lesson taught by a human teacher; it could be a small part of a tutorial agent that can do much more than ask this kind of question; or it could be part of a traditional web-based learning system that also offers smaller interactive elements, such as this conversational module.

Our prior work has shown that learners can learn how to argue with such a scaffold [9]. However, our prior work needed a machine learning engineering process to develop the classifiers behind the adaptation mechanism. Such multistep processes, typically including document annotation, iterative training, and model calibration steps are dominant in existing literature (e.g., [7], [10], and [11]). As an alternative, semantic similarity-based approaches (e.g., a literature review [12]), need no multistep training but rather example input statements (e.g., [13]). Following semantic similarity-based approaches, the workflow proposed by the authors in this article utilizes pretrained transformer-based large language models to minimize the required engineering steps while going beyond classifier performance in comparison with earlier semantic similarity-based approaches. By transformer-based large language models, we refer to a type of deep learning models, which contain encoders and decoders with self-attention capabilities [14]. These models are often pretrained on massive datasets, allowing them to learn a broad understanding of language. These are; however, susceptible to differences in language. Transformer-based large language models now promise to combine the best of both worlds. These models have been analyzed and discussed in other domains, for instance, in bioinformatics [15].

In this work, we are now concerned with whether they indeed make the learning engineering process simple (which they do) while keeping up a reasonable technical and pedagogical quality. Specifically, we evaluated transformer-based large language models as classifiers, which are responsible for identifying the components of Toulmin's model of argument. The evaluation was done through three different test cases to show the performance in different domains.

We propose a systematic workflow to provide content and adaptation mechanisms for a conversational module as described above. The workflow uses transformer-based language models. Like this, the classical machine learning procedures can be avoided, and the learning engineering process becomes viable for a larger group of potential learning engineers, such as teachers, educational technology consultants, or media and content developers. The workflow consists of the following three steps.

1) *Defining the Initial Question*—formulating the starting question for such a conversational module based on existing learning materials.
2) *Defining Expected Phrases*—specifying the input that transformer-based language models need to function as classifiers that decide about the subsequent turns that the conversational module takes.
3) *Defining the Dialogue Structure*—specifying the adaptive dialogue structure, i.e., the turns the classifiers can choose between.

This workflow is systematic, implying that the steps for developing a conversational module remain consistent across diverse learning domains.

This workflow aligns with the educational goals of teaching concepts, definitions, or terminologies through argumentation. Such a conversational module can be integrated, as one element, into a wider conversational interface or educational agent, as described in [16]. In addition, similar to the approach taken in [17], which focuses on structuring and guiding peer interaction with an emphasis on knowledge building, our proposed conversational module can be employed in massive open online courses (MOOCs) to support students and enhance knowledge acquisition by incorporating argumentative conversations through conversational agents. Following Weber et al. [18] taxonomy of educational conversational agents, we understand our conversational agent to be unspecific to different target groups, to support learning factual knowledge and applying it, and thereby to support both practice at these levels, and preparation for subsequent learning phases.

To evaluate the workflow, we created three different conversational modules. For each, we assessed two different quality aspects: 1) the quality of classifying learners' responses based on Toulmin's model of argument using pretrained large language models and 2) the coherency of conversation including the agent follow-up questions to a user response based on classifier results (cf., research questions in Section III).

The rest of this article is organized as follows. In Section II, we elaborate on Toulmin's model of argument [5], computer-mediated environments for argumentation, and argument mining. In Section III, we concretize the research questions that we ask and answer in this work. In Section IV, we describe the workflow in detail. In Section V, we describe the three different conversational modules created as test cases. In Section VI, collecting data and the method used to answer the research questions are described. The results are shown in Section VII. Finally, Section VIII concludes this article.

II. Background and Related Work

A. Toulmin's Model of Argument

The primary factor that has led to the creation of novel evaluation and visualization techniques for argument representation is the need for simple but effective ways to break down, analyze, and eventually better understand arguments. In both scientific and educational settings, one widely recognized framework is Toulmin's argument model [5]. According to the model, an argument comprises six distinct elements. A claim is a statement that requires validation. Evidence is information or knowledge utilized to support the claim. The warrant establishes the logical link between the claim and the evidence. A qualifier is a term or phrase indicating the level of certainty in the claim. The rebuttal addresses aspects where the claim does not hold true, essentially presenting an alternative valid perspective. Finally, the supporting materials that reinforce the warrant are referred to as the backing component. Based on the Toulmin's model of argument, only three components—the claim, warrant, and evidence—are essential for a complete argument, and they are known as the core components.

In general, Toulmin's model is particularly well suited for scientific contexts due to its emphasis on logic, evidence, and reasoning. It helps to analyze and compare the qualities of different arguments based on the presence/absence of the structural components and their interrelations [19]. Based on [20], Toulmin's model is generally simpler and more domain-general than other argument ontologies. This simplicity had the advantage of reducing some types of user errors. In this work, we used Toulmin's model to assess the responses to the argumentative question, which is asked by a conversational agent. The agent then gives feedback based on the identified Toulmin's core components, claims, warrants, and evidence, to help users to fill the argumentative gaps in their answers.

B. Computer-Mediated Environments for Argumentation

Various studies have utilized computer-mediated environments to assess argumentative essays written by students and give feedback to refine their arguments. Afrin et al. [21] developed a web-based intelligent writing assistant, evaluating four interfaces to determine their effectiveness for students. The interfaces differed in feedback types, focusing on unit span (sentence and subsentence) and levels of surface and content revisions. Surface revisions involved grammar and organizational changes, whereas content revisions encompassed meaningful textual changes following Toulmin's model [5], including claims, reasoning, and rebuttals. Their comparison revealed that the most effective interface displayed detailed surface and content revisions at the sentence level.

In [10], a tutoring system (ArgueTutor) was developed to help students to write more convincing essays. The students' task was, first, to read the debate of two teachers on a specific topic, and then, write an argumentative essay. The system was turn based. In each round, first, an argumentative essay should be written by a student, and then, ArgueTutor analyzed the essays and gave feedback using deep learning methods. After each round, the students had the chance to improve the essay based on the received feedback. The feedback received by students constituted a short summary based on the number of argumentative components based on Toulmin's model [5] and a readability score, which is calculated based on [22]. Providing such feedback by the agent enhanced the quality of the essays. By comparing the essays, the authors found that students who received feedback from ArgueTutor wrote more convincing texts and with a better quality of argumentation than the control group.

Configurable Argumentation Support Engine (CASE) was created to assist teachers in tailoring tutorial agents [23]. It enabled teachers or learning engineers to establish patterns for the tutorial agent to analyze student activities. By identifying patterns in student responses or activities, the agent delivered hints and feedback to support argumentation learning activities. Leveraging rule-based pattern-matching techniques, CASE detected pedagogically relevant patterns in argument diagrams and provided feedback and hints to teachers for customizing tutorial agents in terms of argument patterns, tutorial actions, and tutorial strategies.

In this article, we complement such existing work by investigating a conversational module that asks students not for a full argumentative essay. Rather, the goal of the conversational module we design is to engage students in remembering or at least reviewing and repeating a given definition and applying it to a given example, demonstrating these two steps by providing a fully argumentative answer. Computationally, the difference lies both in interaction design (as we do not have a full essay as the object of the learner's activity, and an object around which the interaction of the conversational module and the learner is centered), and in technology design (as the user answers to questions are much shorter than a typical essay.)

C. Argument Mining

Argument mining is a field of research in computational linguistics that aims to automatically identify arguments in unstructured texts [24]. In this process, three distinct subtasks can be identified. The first task involves distinguishing between argumentative sentences and nonargumentative sentences. This

step is considered a binary classification task and has been explored in previous studies (e.g., [25]). The second task is called argument classification or classification of argument schemes [10]. This task focuses on the identification of the type of argumentative sentences based on various argumentative schemes, such as Toulmin's model [5] or claim–premise scheme [26]. Finally, the last task is identifying the relationships among the identified argumentative components [27].

Previous research has approached these tasks differently. Recently, neural network architectures, such as long short-term memory (LSTM) networks and convolutional neural networks, have been commonly employed. More recently, Transformer-based models, such as bidirectional encoder representations from transformers (BERT) [28], based on the Transformer models proposed by Vaswani et al. [14], have also been utilized in this domain. For instance, in [29], the goal was to model the argumentation level of student-written business model pitches by capturing argumentative components (major claims, claims, and premises). To classify the argumentative elements of a given text, they trained and adjusted the hyperparameters of an LSTM model [30]. As training data, they gathered a corpus of 200 student-written business model pitches in German and annotated them based on the argumentative components.

The use of pretrained language models has become popular in the natural language processing community. They convert words or sentences to numerical vector spaces that incorporate contextual information about words or sentences. Dealing with such vector spaces has been a prevalent approach, which emerged due to rapid advances in neural networks. In these vector spaces, words or sentences with similar meanings are positioned closer to each other. This representation, therefore, captures semantic relationships between them, allowing algorithms to understand and work with the contextual meaning of them more efficiently. These word embedding models have outperformed traditional approaches in many natural language processing tasks, especially in argument mining. In the last few years, many different pretrained language models have been created and used that have been leveraged differently in argument mining. For instance, Habernal and Gurevych [7] used word embedding vectors trained on part of the Google News dataset [31] to identify the argumentative components, such as claims and premises. Wambsganss et al. [10] trained a predictive model following BERT architecture to classify text tokens as claim, premise, or nonargumentative following Toulmin's model of argument [5] in students' essays about the topic of "*Does TV make students aggressive?*" To train the model, Wambsganss et al. [32] used a German corpus, which contains 1000 business model peer reviews written by students

Transformer-based language models were also used for converting sentences to numerical vectors. The vectors can be used as features to measure the similarity. Xia et al. [33] employed BERT as a feature extractor to train a series of machine learning models (e.g., logistic regression and random forest) for identifying the argumentative components and relations. As training data, they collected and annotated 1269 sentences including 164 discussion threads covering eight topics, such as abortion and marriage. Abro et al. [34] proposed a framework, which contained two submodels, namely, intent classifier and argument similarity. In the latter, they looked for the most similar argument, which referred to the user's utterances. To produce high-quality sentence representations, which are needed to measure the similarity, the authors combined contextualized word features from the BERT with some additional information, and used cosine similarity to compare the sentence representations with the arguments in the system.

The implementation of machine learning and AI techniques for argument mining relies heavily on the availability of annotated documents, which serve as a training set for predictive models. However, the process of constructing an annotated dataset is a complex and time-consuming endeavor, requiring substantial resources such as expert teams to ensure the acquisition of consistent and homogeneous annotations [35]. Furthermore, it is important to note that different datasets are often created with specific objectives or for particular genres, making them less suitable for all approaches or all stages of the argumentative tasks (see [10] and [29]).

In this article, we present a systematic workflow for creating a conversational module and giving adaptive feedback based on Toulmin's core components. One of the goals of the workflow is to minimize the required engineering, such as collecting huge amounts of data and annotating processes for identifying the core component in various learning domains. To this end, we utilize pretrained large language models sentence BERT (SBERT) [36], [37]. These models convert sentences to numerical vectors that can be compared using cosine similarity. By utilizing SBERT models and collecting a few reference samples for each Toulmin's component, they can be identified using cosine similarity.

To the best of the authors' knowledge, no prior studies have investigated how to have a systematic workflow for developing such adaptive-argumentative dialogues for conversational agents. To fill the gap, our goal has been to create a systematic workflow for creating a conversational module, which consists of a dialogue structure and an adaptation mechanism for argumentative questions on different topics. In other words, by taking the steps of the workflow, we can ask an argumentative question, and create machine learning classifiers with pretrained models to identify Toulmin's core components and define the required branches in the dialogue based on the core components.

D. Research Gaps

Building on the above described prior work, in this work we are interested in the learning engineering process of building, for a specific learning topic, a conversational module that supports learners in developing a full argumentation. This is different from what was done in [38], in which the focus was on the visualization of arguments and collaboration. There are two main reasons why this process is labor intensive.

First is *annotation of training data:* To have an adaptation mechanism that can support argumentation, computational argument-mining techniques are required to analyze and understand argumentation within different learning domains. Conducting argument mining through machine learning and artificial

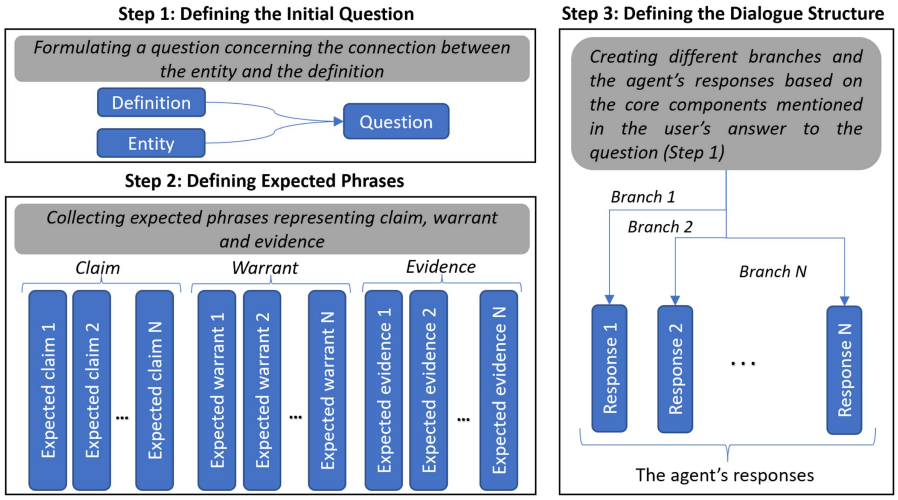

Fig. 2. Outline of the workflow for engaging in an argumentative conversation. The steps correspond to tasks assigned to a learning engineer. The pretrained transformer models handle the classification of learners' responses, utilizing expected phrases (it is not depicted separately in the figure).

intelligence techniques needs the availability of annotated documents, which serve as training sets for predictive models [35]. Creating an annotated dataset is a complex and time-consuming task that often requires substantial resources, including teams of experts, to ensure consistent and homogeneous annotations [39]. For instance, in the work by Lauscher et al. [40], multiple calibration phases were conducted to enhance interannotator agreement, resulting in increased costs. This implies that creating a sufficiently large annotated dataset for new domains becomes impractical, leading to degraded performance in argument structure parsing due to limited data availability [35]. In addition, different datasets are often constructed with specific objectives or for particular genres, making it difficult to find a single dataset that suits all approaches, domains, or stages of argument mining [35]. Subsequently, corpus annotation represents a severe knowledge engineering bottleneck for systems based on computational linguistics, specifically also true for argument mining-based systems [39]. Such annotation was done in much of the above discussed prior work (e.g., [6], [7], and [11]). Transformer-based large language models are pretrained, on the other hand, and in this work, we investigate the performance achievable out-of-the-box with them.

Second is *content development:* Like any educational system, conversational agents or interfaces require educational content. Similar to other adaptive systems, this goes beyond the primary content and includes all kinds of feedback or responses. In the case of educational conversational agents, this corresponds to the adaptive dialogue design. In this article, we investigate a systematic for building an adaptive dialogue structure for one particular type of conversation.

Finally, learning engineering has a particular challenge at the intersection of technical and pedagogical designs. Technological capability and pedagogical design need to be well aligned with each other. Subsequently, we investigate a workflow in which technical and pedagogical developments of the conversational module are treated as tightly interrelated to the extent of being part of a single design workflow.

Learning and technologies literature contain prior work on conversational educational agents, and there exists a substantial body of literature on argument mining techniques. However, literature on the learning engineering process of conversational educational agents is extremely scarce, to the extent that we identified only widely related literature on authoring for adaptive educational systems, such as [41], [42], [43].

III. Research Questions

The workflow we investigate in this article has three main steps, as shown in Fig. 2. Following the steps, we can create a conversational module containing a starting question based on existing learning materials, analyzing the learners' answers using transformer-based language models, and specifying the adaptive dialogue structure. We ask the following two research questions that target two different aspects of the quality of the conversational modules.

1) *RQ1 (classifier quality):* To what extent can pretrained language models accurately recognize the specified Toulmin's model components (claim, warrant, and evidence) when provided with manually predefined example input statements?
2) *RQ2 (conversational coherence):* How coherent are the follow-up questions generated by the agent in response to a user's input, considering the classifier outcomes and the systematically defined adaptive dialogue structure?

Regarding RQ1, we use the F1-macro score as a measure of the classifier quality. RQ2 is based on the concept of "conversational coherence." This concept refers to the quality of turns in a dialogue to reasonably follow up on each other [44]. In other works, this concept has been used to measure the quality of a user–agent conversation [9], [45], as a factor that impacts the user experience of interaction with a conversational agent. In this work, we assess whether the next follow-up question of the agent is coherent (details in Section VI). Note that having a reasonable performance in the classifier quality is a prerequisite

for having coherent conversations. However, ideally, dialogue design can serve as a cover-up for misclassifications.

In order to evaluate the workflow, first, we define three test cases (see Section V). For each test case, we apply the systematic workflow in order to develop a conversational agent module. Second, we collect data in each of the test cases in order to answer both RQ1 and RQ2 (see Section VI).

IV. Systematic Workflow

The goal is to develop a conversational agent module, in which the learner is asked to develop an argumentative answer to a question. They are also guided by the conversational agent toward developing a full argument if argumentative parts are missing (cf., Fig. 1). We assume that individuals responsible for creating the conversational module possess a combination of learning domain knowledge and technical expertise. This could be a single learning engineer proficient in both instructional and technical domains, or a collaborative team comprising an instructor and an engineer. For simplicity, below we always write "the learning engineer."

The workflow consists of three main steps. "*Step 1: Defining the Initial Question*"—the learning engineer formulates the starting question for such a conversational module based on existing learning materials (see Section IV-A). "*Step 2: Defining Expected Phrases*"—the learning engineer needs to specify the input or the expected answers that transformer-based language models need to function as classifiers that decide about the subsequent turns that the conversational module takes (see Section IV-B). The output of the second step contains example phrases that represent reasonable claim, warrant, and evidence (following Toulmin's model of argumentation) to the agent's question defined in the first step. In the third step, "*Step 3: Defining the Dialogue Structure,*" the learning engineer specifies the adaptive dialogue structure, i.e., the turns the classifiers can choose between. It includes follow-up questions in cases where the learner's response misses core elements of Toulmin's model, and thus, does not provide a fully developed answer to the agent's question (see Section IV-C). Fig. 2 summarizes the workflow.

A. Step 1: Defining the Initial Question

The initial question of the conversational module consists of two main parts: 1) a definition or term mentioned in learning materials and 2) an entity or a specific example. Given the definition and the example, the question should ask the learner to explain why/why not the given example fulfills the definition or not. Table I gives examples of such questions, including a definition and a given example, in various learning domains. For instance, in the topic of biology, the definitions of the animal types could be as follows: *the mammal is a type of animal with warm blood and a hairy body. Examples of mammals are cows and elephants. Reptiles are types of animals with cold blood and dry scaly skin. Examples of reptiles are snakes and crocodiles*. Given the definition, the learners are asked which definition can be fulfilled by lions as an example. All the required information to answer the question can be provided by the conversational agent during the conversation or by teachers beforehand. Depending on the learning domain, the example is very short (e.g., "a lion," or "Jupiter" in Table I) or is longer (e.g., an elaborate scenario for which applicability of the GDPR should be decided (ibid). Different methods can be used by the learning engineer to support argumentation, such as providing extra direction or prompting questions [46]. The inclusion of "*Explain why?*" or similar phrases can be used to ask learners to construct an argument and to justify their claim using the provided definitions [46].

TABLE I
Example Topics, Definitions, and Argumentative Questions

Topic	Definition	Example question
Biology	The definitions of five types of animals: mammals, reptiles, birds, amphibians, insects	Is *a lion* a mammal or not? Explain Why?
Astronomy	The definition of a planet	Is *Jupiter* a planet or not? Explain Why?
Intelligence	The definition of intelligence	Is *a monkey* intelligent or not? Explain Why?
Physics	The definition of the state of matter	Is *a bottle of milk* solid, liquid or gas? Explain Why?
The GDPR	The definition of the GDPR principles	Based on *the scenario*, Which one of the seven principles has been violated? Explain Why?

B. Step 2: Defining Expected Phrases

We view each user's response as an argument, employing Toulmin's framework. In the responses, the direct response serves as the claim. The relevant information related to the entity (or the example) is regarded as evidence, while the segment of information within the definitions that establishes a connection between the evidence and the claim represents the warrant component.

The diversity of the expected phrases determines the number of branches in the dialogue and the agent's feedback. For instance, in this question "*What is the state of matter (solid, liquid, or gas) of honey?,*" if the learning engineer aims to assess only the correctness of the learners' answers, the expected phrases of the claim component should cover the correct answer, such as "*Honey's state of matter is liquid*." However, to offer specific feedback on common errors and misconceptions, the learning engineer needs to provide expected phrases that address these typical mistakes such as "*I think honey should be solid*."

Besides the claim component, the learning engineer should define the expected phrases of warrant and evidence component. In this example, the expected phrases of the warrant consist of the phrases that cover the key points within the definition of each state of matter, such as "*particles roll over each other and settle on the bottom.*" The expected phrases of evidence include the relevant information about the honey (used as an example by the learning engineer in the first step of the workflow) to support the claim, such as "*it takes the shape of the container.*" In the case of using a scenario to describe an entity, the learning engineer restricts the number of expected phrases of evidence to

the pieces of information mentioned in the description that can support the claim.

To give an adaptive feedback, the learning engineer generates representative phrases for claim, warrant, and evidence. These are used as inputs for pretrained transformer-based language models in the conventional module. If the learners' responses lack sufficient reasoning, such as a missing claim, warrant, or evidence, the conversational agent asks follow-up questions to support the learners in filling in the missing argumentative components in their answers.

C. Step 3: Defining the Dialogue Structure

In this step, the learning engineer first defines the number of branches based on the expected phrases for each core component, and then, defines the agent's responses or the follow-up questions for each branch, A complete answer should consist of all three core components. By focusing only on the existence of each core component in the learners' answers, the learning engineer is capable of covering eight different branches as follows: "*with_claim*" and "*without_claim*" for the claim component, "*with_warrant*" and "*without_warrant*" for the warrant component, and "*with_evidence*" and "*without_evidence*" for the evidence component.

As mentioned in the previous step, the conversational module can be more adaptive by having specific branches, which deal with the wrong claims or common mistakes of learners. In this case, a ternary value is needed for the claim component, "*correct_claim*" for the correct answers, "*incorrect_claim*" for incorrect answers, such as "*honey is solid,*" and "*without_claim*" for answers, in which the claim is missing. Having a ternary value for the claim and two binary values for the warrant and the evidence, 12 different branches can be generated. The 12 branches can be reduced to only six branches because providing feedback on the warrant and evidence when the claim is incorrect or missing is not reasonable and may cause misunderstandings. In other words, the agent's feedback should address first the missing or incorrect claim component, as the other two components are meant to support the correct claim. Therefore, the initial 12 branches can be reduced to six branches, as outlined below:

1) *incorrect_claim;*
2) *without_claim;*
3) *correct_claim, with_warrant, with_evidence;*
4) *correct_claim, with_warrant, without_evidence;*
5) *correct_claim, without_warrant, with_evidence;*
6) *correct_claim, without_warrant, without_evidence.*

After determining the branches, the learning engineer needs to define the agent's feedback and/or the follow-up questions for each branch. For example, a possible answer to the question about the state of matter of honey might be this: "*I think honey is liquid because it can flow and takes the shape of its container.*" It includes a claim ("*I think honey is liquid*") and evidence ("*it can flow and takes the shape of its container*"), and the warrant component that establishes the connection between the claim and evidence is missing. Based on the different values of expected phrases for each component, the corresponding branch to such answer is "*correct_claim, without_warrant, with_evidence.*" As the agent's goal is to support learners to write a complete answer, a possible agent response could be: "*Great! I understood already a lot of what you are saying—one thing is not clear yet. I do not understand based on which part of the definition of states of matter, you think that honey is liquid. Read the definitions again and tell me which part of it supports your claim.*"

V. Test Cases

In this section, we describe the three different test cases that we created following the steps of the workflow. The summary of the test cases are shown in Table II.

A. Test Case 1

Test Case 1 concerns the European GDPR. The GDPR introduces new definitions and frameworks for the handling and management of personal data. As a result, organizations are required to adapt to the concepts outlined in the GDPR. This topic holds significance for a wide range of professions, making it a typical subject covered in MOOCs at an introductory level. By addressing the GDPR in our test case, we aimed to assess the applicability and effectiveness of our systematic workflow in a context that is relevant and valuable for various professional domains.

1) Defining the Initial Question: To define the initial question, we used the seven GDPR principles (lawfulness, fairness and transparency, purpose limitation, data minimization, accuracy, storage limitation, integrity and confidentiality and accountability—cf., Table III). In addition to the definitions, we needed an entity or an example to which these definitions could be applied. We defined example scenarios based on existing learning materials around GDPR. The example scenario used in our evaluation is as follows.

Question 1: A recruitment agency is located in the EU. The agency places workers in a variety of jobs. Some of the jobs are office jobs and the other jobs are hazardous for which they need to ask specific questions about health conditions, such as the blood group. The applicants already know which type of job they want to apply for. To collect the data of all workers, the agency sends all applicants a general questionnaire, which also includes health-related questions that are only relevant to hazardous occupations. Which one of the seven principles has been violated by the recruitment agency? And why? Explain which principle, which part of its description, and what the agency did exactly, which led to the violation of the principle.

Based on the scenario, the correct answer or the violated principle is the data minimization principle (cf., Table III).

2) Defining Expected Phrases: In this step, we created a list of expected phrases for each component. Considering the question, learners can select any of the seven GDPR principles as their claim. To cover all principles, we added seven statements with the pattern of "*X has been violated*" to the expected list of claims. Here, "*X*" represents the name of each principle, such as "*data minimization principle has been violated.*" Due to the space limit, we moved all the expected phrases for all test cases in the Supplementary Material.

TABLE II
SUMMARY OF THE TEST CASES

Test cases	Question (Step 1)	# of expected phrases (Step 2)	# of branches in the dialogue structure (Step 3)
TC1	Based on the scenario, which one of the seven principles has been violated by the recruitment agency? And why?	38	6
TC2	Based on the scenario, which one of the seven principles has been violated by the hospital? And why?	31	6
TC3	Based on the definitions, do you think monkeys are intelligent or not? And why?	78	5

TABLE III
DEFINITIONS OF GDPR'S PRINCIPLES

The definition of GDPR's principles
Data minimisation: As the third principle, we need to refer to data minimisation. It states that personal data must be adequate, relevant and limited to what is necessary concerning the purposes for which they are processed. Essentially, it means that data can be collected or processed only to achieve the above-mentioned purposes.
Accuracy: Accuracy is the fourth principle meaning that it is required to ensure that personal data are accurate and are kept up to date where it is necessary. The data should also accurately reflect the order of events. Inaccurate personal data – considering the purposes for their processing – must be deleted or rectified without any delay.

Regarding the creation of the expected list of warrants, our focus was primarily on the definition of the violated principle. The list consisted of six statements, including examples, such as "*personal data must be adequate, relevant, and limited to what is necessary in relation to the purposes.*"

Given the scenario describing the imaginary entity (the recruitment agency), the evidence component of users' answers should consist of statements or phrases extracted from the scenario that highlight the factors leading to the violation. We included 13 statements in the expected list of evidence such as "*the blood groups were not related to all applicants.*"

3) Defining the Dialogue Structure: In this step, we defined the dialogue branches and the agent's responses for each branch. As explained in Section IV-C, these responses should assist users in mentioning all the essential components and contribute to a natural and coherent conversation. For example, in case of missing evidence in the user's response, the corresponding branch would be "*correct_claim, with_warrant, without_evidence*". In this branch, the agent's feedback would be: "*Great! I think I understood already a lot of what you are saying—one thing is not clear yet. I do not understand how the agency violated the principle. Read the scenario again and tell me what exactly they did that led to the violation of the principle.*" The feedback starts with a positive statement to acknowledge that the claim was correctly mentioned, followed by a request for the missing evidence component. Similar feedback statements have been crafted for the other branches as well. These statements aim to assist users in addressing any structural gaps in their answers and maintaining a coherent and meaningful conversation. All the agent's responses were listed in the Supplementary Material.

B. Test Case 2

1) Defining the Initial Question: For the second test case, we continued with the topic of GDPR and used a different scenario in which another principle was violated. In this particular example, the principle that has been violated is the accuracy principle, which is explained in Table III. The question and the scenario are as follows.

Question 2: "In a hospital, for each patient, only the last diagnosis of a medical condition continues to be held and the previous diagnoses are deleted. Now, Bob who has been in the hospital for six months wants to know why his treatments have been changed monthly. The hospital cannot answer his question because they just keep the last diagnosis of each patient and delete the old ones. Which one of the seven principles has been violated by the recruitment agency? And why? Explain which principle, which part of its description, and what the agency did exactly, which led to the violation of the principle.

2) Defining Expected Phrases: Similar to the question in Test Case 1, an answer needs to have a claim, which refers to one of the GDPR principles violated in the scenario by the entity (the hospital). The expected phrases for the claim component were identical to those created for Test Case 1. However, the correct or violated claim for Test Case 2 was the accuracy principle. The expected warrants included phrases in the violated principle's definitions, such as "*the data should also accurately reflect the order of events.*" Similarly, the expected evidence for this question consisted of nine statements mentioned in the scenario, which led to the violation of the accuracy principle, for instance, "*only the last diagnosis of a medical condition continues to be held.*"

3) Defining the Dialogue Structure: In Test Case 2, we utilized the same branches defined for Test Case 1. However, in Test Case 2, the "*correct_claim*" referred to the accuracy principle. For each branch, a suitable response for the agent was defined. The agent's responses should help learners to mention all the required components in their answers, and also, the agent's responses should keep the conversation coherent. For example, in the "*without_claim*" branch in which the claim component is missing, the agent would respond with: "*Mmmm, could you clearly specify which principle was violated? And explain why?*" Since the claim component is the most crucial one, the agent should first focus on obtaining the claim if it is missing, regardless of the status of the other components.

C. Test Case 3

1) Defining the Initial Question: In the third test case, we explored the topic of intelligence, which has been previously addressed in our previous works [6], [9]. The selection of this topic was based on the aim of enhancing AI literacy [47]. We focused on the definitions of intelligence and utilized five

definitions of intelligence: acting rationally, acting humanly, thinking rationally, thinking humanly, and the ability to learn from experiences. The first four definitions have influenced various directions of AI research [48]. The fifth definition aligns more closely with the understanding of learning in psychology and learning sciences. As for the entity, we used monkeys, and the question for this test case is as follows.

Question 3: Based on the definitions, do you think monkeys are intelligent or not? Explain why?

2) Defining Expected Phrases: To generate the expected statements for each core component, we took a different approach than in Test Cases 1 and 2. We leveraged a dataset that was previously compiled in [6]. This dataset is publicly available and contains annotated answers for the question specified in Test Case 3. The dataset encompasses 1337 answers related to 12 different entities. From the 155 available answers about monkeys, we randomly selected 20 answers and utilized them to form the expected phrases for each component.

The resulting list of expected claims, warrants, and evidence consists of 20, 35, and 39 statements, respectively. For example, an expected phrase for the claim could be "*monkeys are intelligent,*" for the warrant it could be "*it can be observed thinking and behaving like a human,*" and for the evidence, it could be "*monkeys have acquired the skills to use basic tools*."

3) Defining the Dialogue Structure: In Test Case 3, we employed similar branches to those defined in Test Cases 1 and 2, with two modifications. First, we removed the "*incorrect_claim*" branch, as there was no specific correct answer to the question. Given different definitions, the claim could be different. Second, we renamed the "*correct_claim*" branch to "*with_claim.*" Since there is no specific correct answer (claim), the focus should be on the presence or absence of a claim rather than its correctness. Therefore, we defined only five branches, one related to the answers without a claim, and four more branches for answers with a claim and different values of warrant and evidence.

Once the branches have been finalized, it is important to specify a meaningful response for each branch. For example, in this branch ("*with_claim, with_warrant, without_evidence*"), we defined the agent's feedback as "*Interesting! But the evidence part is missing! Could you explain why you think that monkeys are intelligent based on the definition(s) that you mentioned?*" This feedback aims to prompt the user to provide additional information to support their claim, highlighting the importance of including evidence in their response.

VI. Evaluation Methodology

1) Apparatus: The three abovedescribed test cases were operationalized as web-based, interactive conversational agents using the Bazaar framework[1] [49], [50], and inserting the classifiers developed via our systematic workflow. All the experiments were run on a computer with an *Intel i7 (11800H)* processor running at 2.3 MHz using 32 GB of RAM, running on Windows 11. The most time-consuming part of the computational time was related to downloading and loading the SBERT models, which should be done once for each model. The details of each model can be found on the SBERT official website.[2]

2) Data Collection: We collected two different datasets (Datasets 1 and 2) for each test case, which contained the sample answers to the initial questions. To collect data for Dataset 1, we asked our research team ($n = 7$) to answer the initial question related to each test case. All the answers were split into sentences and, overall, 89 sentences were collected, 31 sentences belonged to Test Case 1, 32 sentences belonged to Test Case 2, and 26 sentences belonged to Test Case 3. Dataset 1 was utilized to compare pretrained transformer-based language models (see Section VI-A).

We also asked the same questions to 100 unique MTurk workers to create Dataset 2 for Test Cases 1 and 2. For Test Case 3, we used parts of the data collected in [6]. This dataset contains 156 answers to questions that correspond to Test Case 3. The rest of the overall 1335 answers in the full dataset (ibid) are answers to questions that concern different examples. We used 136 out of 156 answers to create Dataset 2 and the rest (20 out of 156) was used to create the expected phrases for Test Case 3. All the answers were split into sentences and overall 700 sentences were collected, 199 sentences for Test Case 1, 209 sentences for Test Case 2, and 292 sentences for Test Case 3. Note that *this dataset was only used to evaluate the workflow, not to design the conversational modules*. The dataset will be freely accessible in the Supplementary Material.

3) Annotation of Datasets and Interrater Agreement: Both Datasets 1 and 2 were coded for Toulmin's model's core elements, except for the subset related to Test Case 3, which was already annotated. New annotations done for this publication were done by two annotators (the first author included) and they coded the datasets used in Test Cases 1 and 2. Each annotation stated whether the three relevant Toulmin's model elements (claim, warrant, and evidence) are present in a user statement. In addition, the correctness of claim components for Test Cases 1 and 2 was also considered.

The annotation process consisted of three steps. First, the annotators engaged in discussions to establish a consensus on the definition of each component (claims, warrants, and evidence). This step ensured a shared understanding among the annotators. Second, 50 sentences from Dataset 2 (25 from each test case) were randomly selected and independently annotated by the two annotators to assess interrater agreement. Finally, any disagreements that arose during the second step were discussed, and the remaining data were annotated by the first author. Cohen's kappa (κ) value was used to quantify the interrater agreement, taking into account the possibility of agreement occurring by chance. This statistical measure helped assess the level of agreement between the annotators. The κ values for the claim, warrant, and evidence were 0.90, 0.62, and 0.71 respectively. The κ value for the claim was almost perfect, however, for the warrant and evidence, we achieved a substantial agreement. Table IV represents the collected data for all three test cases based on the number of each argumentative component.

[1] [Online] Available: https://github.com/DANCEcollaborative/bazaar

[2] [Online] Available: https://www.sbert.net/docs/pretrained_models.html

TABLE IV
NUMBER OF COMPONENTS IN EACH DATASET

Label	Test Case 1		Test Case 2		Test Case 3	
	Dataset 1	Dataset 2	Dataset 1	Dataset 2	Dataset 1	Dataset 2
# of *correct_claim*	10	33	9	13	5	139
# of *incorrect_claim*	5	66	5	83		
# of *without_claim*	16	100	18	113	21	153
# of *with_warrant*	8	15	5	10	8	111
# of *without_warrant*	23	184	27	199	18	181
# of *with_evidence*	9	38	8	54	15	119
# of *without_evidence*	22	161	24	155	11	173

In test case 3, we used one label, "*with_claim*", for both "*correct_claim*" and "*incorrect_claim*" labels.

A. RQ1 (Classifier Quality): Comparing Pretrained Language Models

We assumed that a pretrained large language model is chosen by the learning engineer based on some decision criteria (e.g., performance versus available hardware, pricing, existing models that are already used within the socio-technical system of the learning engineer, etc.). We also assumed that the choice might matter a bit but not substantially. We tested this assumption informally for each test case and the three core components separately. To this end, for each test case and each component, we compared 16 different pretrained language models (SBERT models), such as *all-mpnet-base-v2* and *paraphrase-multilingual-mpnet-base-v2* [37]. In addition to SBERT models, various other models and architectures can be employed. However, to minimize engineering efforts and model complexity, we opted for SBERT models, which come pretrained, tuned, and well documented, aligning with one of the workflow's objectives. Analyzing the details of the models in terms of how they were trained and their structure was not in the scope of the current study. The list of all models and more details are described in the Supplementary Material.

The selection of the best model involved comparing different models with a range of similarity thresholds based on the F1 score. A similarity threshold needed to be defined for each model by which the agent can decide about the final label of the users' statements. To this end, we defined a range of thresholds from 0.3 to 0.95 with a 0.05 increment. For each model and its threshold value, we used the expected phrases to identify the components in Dataset 1. For components with ternary values (e.g., the claim component in Test Cases 1 and 2, which includes "*correct_claim,*""*incorrect_claim*," and "*without_claim*"), we used the F1 macro score. However, for binary components (e.g., warrants and evidences in all test cases, and claims in Test Case 3), we compared models based on the F1 score of the main class, such as "*with_warrant*," "*with_evidence*," and "*with_claim*."

B. RQ1 (Classifier Quality) on Dataset 2

Following the above comparison of available transformer-based large language models, we chose the best-performing model and the similarity threshold value for each test case and each component. Then, we assessed the performance of the selected model on Dataset 2. We chose the best-performing model as the large language model performance, in general, is still on the rise, so the resulting performance on Dataset 2 would still not be overly optimistic as an outlook for the future.

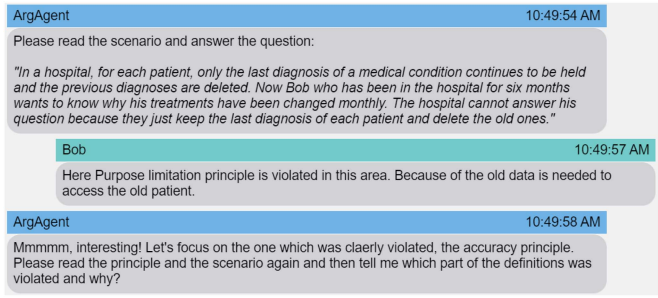

Fig. 3. Coherent response from the agent belongs to Test Case 2.

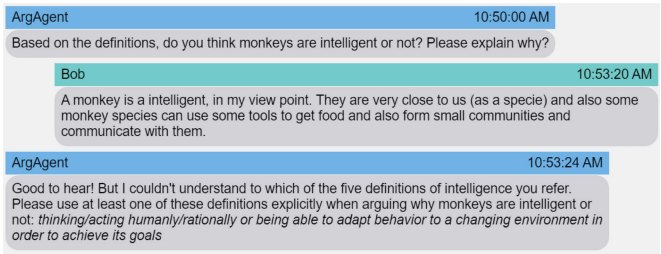

Fig. 4. Coherent response from the agent belongs to Test Case 3.

To assess the performance of the best model on Dataset 2, we utilized precision, recall, F1 score, and accuracy. In addition, we employed the macro and weighted F1 scores.

C. RQ2 (Conversational Coherence)

In order to assess conversational coherence (RQ2), we coded the sequence of "the initial agent question—user response—follow-up agent question" for each test case, separately. The agent selected the follow-up questions or responses (defined by the learning engineer in the third step of the workflow) based on the classifier results from the selected/best classifier according to results from RQ1. The whole coding process was done by the first author. As a result, we give the percentage of coherent sequences in the overall number of statements in Dataset 2 ($n = 335$).

Figs. 3 and 4 show sequences that were coded as coherent. Fig. 3, from Test Case 2, shows how the agent's response can be coherent even though a misclassification happened. The agent classifies the second user's sentence as evidence, which is incorrect. Fig. 4, from Test Case 3, shows the agent asking the

TABLE V
MEAN (M) AND STANDARD DEVIATION (SD) OF F1 SCORES OF ALL PRETRAINED MODELS ON DATASET 1 BASED ON EACH COMPONENT

Dataset 1	Claim	Warrant	Evidence
Test Case 1	$M = 0.78$ $(SD = 0.07)$	$M = 0.76$ $(SD = 0.05)$	$M = 0.86$ $(SD = 0.08)$
Test Case 2	$M = 0.74$ $(SD = 0.04)$	$M = 0.88$ $(SD = 0.08)$	$M = 0.80$ $(SD = 0.07)$
Test Case 3	$M = 0.97$ $(SD = 0.03)$	$M = 0.88$ $(SD = 0.06)$	$M = 0.69$ $(SD = 0.07)$

TABLE VI
BEST MODEL AND ITS THRESHOLD (TH.) FOR EACH CORE COMPONENT (CO.) BASED ON DATASET 1

Test Case 1			
Comnent	**Model**	**Th.**	**F1 score**
Claim	*msmarco-MiniLM-L12-cos-v5*	0.35	0.92
Warrant	*distiluse-base-multilingual-cased-v1*	0.50	0.87
Evidence	*all-mpnet-base-v2*	0.50	0.96
Test Case 2			
Claim	*paraphrase-multilingual-mpnet-base-v2*	0.55	0.84
Warrant	*all-distilroberta-v1*	0.60	0.95
Evidence	*paraphrase-MiniLM-L3-v2*	0.50	0.92
Test Case 3			
Claim	*all-distilroberta-v1*	0.50	0.94
Warrant	*msmarco-MiniLM-L6-cos-v5*	0.60	0.95
Evidence	*all-distilroberta-v1*	0.50	0.81

TABLE VII
TEST CASE 1: THE RESULTS OF THE SELECTED MODELS FOR QUESTION 1 ON DATASET 2

Claim component			
Model	*msmarco-MiniLM-L12-cos-v5*		
Label	**Precision**	**Recall**	**F1 score**
incorrect_claim	0.74	0.91	0.82
without_claim	0.94	0.77	0.85
correct_claim	0.89	0.97	0.93
accuracy		0.85	
macro avg.	0.86	0.88	0.86
weighted avg.	0.86	0.85	0.85
Warrant component			
Model	*distiluse-base-multilingual-cased-v1*		
Label	**Precision**	**Recall**	**F1 score**
without_warrant	0.98	0.95	0.97
with_warrant	0.57	0.80	0.67
accuracy		0.94	
macro avg.	0.78	0.88	0.82
weighted avg.	0.95	0.94	0.94
Evidence component			
Model	*all-mpnet-base-v2*		
Label	**Precision**	**Recall**	**F1 score**
without_evidence	0.96	0.80	0.87
with_evidence	0.50	0.87	0.63
accuracy		0.81	
macro avg.	0.73	0.83	0.75
weighted avg.	0.87	0.81	0.83

follow-up questions based on having identified the warrant as the missing core component.

VII. RESULTS

As it stands, the workflow was carried out, for each test case, once. Based on this experience, each test case took around a few hours to create a conversational module in a new learning domain. It is under the assumption that an operative large language model is available and selected, and the learning engineer has learning materials in traditional formats at hand (i.e., only needs to choose between definitions, and examples, not author them from scratch).

A. RQ1 (Classifier Quality): Comparing Pretrained Language Models

In each test case, we compared the F1 score of all 16 SBERT models on Dataset 1. The average performance of the models for each test case and component is given in Table V.

In Test Cases 1 and 2, we selected the best model for each core component based on the F1 scores. Notably, for Test Case 3, we selected the second-best models for claims and warrants, as all metrics for the best model were 1.00, which could indicate a potential bias toward Dataset 1. Table VI gives the best models and their associated similarity threshold for each core component. For instance, in Test Case 3, using the pretrained model of *all-distilroberta-v1* with a similarity threshold of 0.5 resulted in the F1 score of 0.94 in identifying evidence on Dataset 1. Overall, we note that differences between models are not substantial, which supports our assumption that model comparison and selection need not necessarily be part of the learning engineering process.

B. RQ1 (Classifier Quality) on Dataset 2

The best model for identifying the claim component in Test Case 1 was *msmarco-MiniLM-L12-cos-v5*, which achieved the highest F1 score (0.92) among the other models on Dataset 1. When applied to Dataset 2, this model achieved precision, recall, and F1 score values of 0.74, 0.91, and 0.82, respectively, for identifying incorrect claims. In Test Case 1, the macro average F1 scores of all the components on Dataset 2 ranged from 0.75 to 0.86, which is a promising result. Table VII displays the results of identifying the core components in Test Case 1 on Dataset 2.

In Test Case 2, *paraphrase-multilingual-mpnet-base-v2* achieved the best F1 score on identifying claims in Dataset 1. By applying the model on Dataset 2, the model achieved precision, recall, and F1 score values of 0.87, 0.73, and 0.80, respectively, for identifying incorrect claims. Similar to Test Case 1, the results were encouraging. The macro average of F1 scores of all the components on Dataset 2 was between 0.76 and 0.83. In Table VIII, the results related to Test Case 2 were given.

The best model in Test Case 3 was *all-distilroberta-v1*, which achieved the highest score on Dataset 1. Using the model on Dataset 2, it achieved the macro average F1 scores of 0.77, 0.74, and 0.66 for claims, warrants, and evidence, respectively. The results for Test Case 3 are given in Table IX. By analyzing the performance of the classification of the core components in three different test cases, we showed how well the components can be identified in conversational modules created based on the workflow.

When we compare these results to results from our own prior work, they are extremely encouraging. Mirzababaei and Pammer-Schindler [6] developed random forest classifiers based on a training dataset with 1337 annotated user statements and achieved F1 scores of 0.77, 0.88, and 0.71 for claims, warrants,

TABLE VIII
TEST CASE 2: THE RESULTS OF THE SELECTED MODELS FOR QUESTION 2 ON DATASET 2

Claim component			
Model	*paraphrase-multilingual-mpnet-base-v2*		
Label	**Precision**	**Recall**	**F1 score**
incorrect_claim	0.87	0.73	0.80
without_claim	0.79	0.91	0.85
correct_claim	0.78	0.54	0.64
accuracy		0.82	
macro avg.	0.81	0.73	0.76
weighted avg.	0.82	0.82	0.84
Warrant component			
Model	*all-distilroberta-v1*		
Label	**Precision**	**Recall**	**F1 score**
without_warrant	0.98	0.99	0.98
with_warrant	0.75	0.60	0.67
accuracy		0.97	
macro avg.	0.87	0.79	0.83
weighted avg.	0.97	0.97	0.97
Evidence component			
Model	*paraphrase-MiniLM-L3-v2*		
Label	**Precision**	**Recall**	**F1 score**
without_evidence	0.94	0.86	0.90
with_evidence	0.69	0.85	0.76
accuracy		0.86	
macro avg.	0.82	0.86	0.83
weighted avg.	0.88	0.86	0.87

TABLE IX
TEST CASE 3: THE RESULTS OF THE SELECTED MODELS FOR QUESTION 3 ON DATASET 2

Claim component			
Model	*all-distilroberta-v1*		
Label	**Precision**	**Recall**	**F1 score**
without_claim	0.90	0.64	0.75
with_claim	0.70	0.92	0.80
accuracy		0.77	
macro avg.	0.80	0.78	0.77
weighted avg.	0.80	0.77	0.77
Warrant component			
Model	*msmarco-MiniLM-L6-cos-v5*		
Label	**Precision**	**Recall**	**F1 score**
without_warrant	0.76	0.94	0.84
with_warrant	0.85	0.51	0.64
accuracy		0.78	
macro avg.	0.81	0.73	0.74
weighted avg.	0.79	0.78	0.77
Evidence component			
Model	*all-distilroberta-v1*		
Label	**Precision**	**Recall**	**F1 score**
without_evidence	0.73	0.71	0.72
with_evidence	0.59	0.61	0.60
accuracy		0.67	
macro avg.	0.66	0.66	0.66
weighted avg.	0.67	0.67	0.67

and evidence, respectively. Also, in further related work in argumentation mining, the F1 scores we achieve in the here described work are absolutely comparable (e.g., F1 scores in the ranges of 60%–90% in [7], [10], [51], [52], and [53]).

These findings indicate that the systematic workflow, with its focus on leveraging pretrained language models and minimizing engineering efforts, can yield comparable performance to more complex models trained with specific data and fine-tuning. This demonstrates the potential of the workflow to streamline the development process and reduce the required engineering work to have such a conversational module in various learning domains.

C. RQ2: How Coherent are Agent Follow-Up Questions to a User Response Based on Classifier Results?

In this section, we address RQ2, which focuses on the coherence of conversations produced through our systematic workflow. For each test case, we calculated the percentage of coherent dialogue turns including the agent's initial question, the user's response, and the agent's follow-up question or response. The percentages of coherent conversations in Test Cases 1–3 are 84%, 79%, and 80%, respectively.

VIII. CONCLUSION

Our work has been motivated by the potential of modern conversational agents to support learning to argue, and learning through argumentation on the one hand, and the technology of transformer-based language models on the other hand. The latter holds the promise to alleviate an acknowledged bottleneck in the creation of language-based adaptive systems: creating substantial and domain-specific datasets (data collection and annotation). This bottleneck can be addressed with transformer-based language models, as they allow training classifiers only with very few training examples ("example phrases" in Step 2 of the workflow we presented, cf., Fig. 2). This bottleneck comes on top of the challenge in adaptive educational systems to align well technology and (interactive) content.

In this work, we investigate the promise of transformer-based language models by proposing and investigating a systematic learning engineering process for a conversational module that can do the following:

1) provide a concept or a definition from the student's learning domain;
2) ask the learner to apply this definition to an example;
3) give adaptive feedback to the learners on their reasoning.

Our findings show that the average of F1 scores of classifying Toulmin's core components—claim, warrant, and evidence—in all three test cases was $M = 0.79$ $(\text{SD} = 0.06)$. As described in the results in Section VII-B, these results are absolutely comparable with the performance reported in prior literature *without necessitating the collection and annotation of training data*. Further, our findings show that the percentage of coherent dialogue turns was 84%, 79%, and 80% for Test Cases 1– respectively.

These findings have implications for the development of educational technology and teaching practice that makes use of educational technology. First, for the development of educational technology, the results emphasise the boost that transformer-based language models give to adaptive learning technology, such as conversational modules. Without the necessity to go through the process of engineering a machine-learning-based system, a very reasonable performance of the overall educational system can be achieved. As transformer-based models continue to advance and improve, their benefits for learning engineering are expected to increase. These models offer powerful natural

language processing capabilities, enabling more sophisticated and context-aware interactions between conversational agents and learners.

Second, the study highlights the importance of considering both classifier or adaptation mechanism quality and dialogue coherence when evaluating the performance of conversational agents. While the quality of the classifier or adaptation mechanism is crucial for accurate understanding and response generation, the resulting dialogue coherence plays a vital role in ensuring meaningful and engaging interactions with learners. This finding suggests that future research and development efforts should focus on optimizing both aspects to enhance the overall effectiveness of educational conversational agents. For the development of educational technology, this means a shift from technology development toward learning engineering, in which technical competencies are important when developing educational technology, but major efforts required are knowledge engineering efforts that require domain and didactical domain knowledge. This is what is generally understood as "learning engineering" then.

As the development of adaptive conversational modules becomes less of a technical, and more of a domain and didactical effort, creating specific conversational modules becomes realistic for individual educators. This will be an important shift in teaching practice if educators can easily handle the learning-engineering required for their own conversational modules.

From a pedagogical perspective, asking argumentative questions and providing argumentative feedback on learners' responses not only leads to learning to argue but also arguing to learn. By leveraging transformer-based language models as used in the proposed workflow, argumentation can be injected into various learning domains. Using transformer-based language models enables us to reduce the time and resources traditionally required for developing argumentative conversational modules, making it a practical and efficient solution for modern classrooms. Furthermore, teachers can leverage conversational modules as dynamic tools for presenting concepts, prompting students to apply knowledge, and providing tailored feedback on their reasoning.

A main direction for future research following on this work of ours, as well as in connection to other ongoing research, is to carry out more user-oriented research. By this we mean, to evaluate a workflow, such as ours, in user studies with people who would act as learning engineers (educators, or support staff close to them) who could realize this "ad hoc creation" of adaptive learning technology. While our proposed workflow has the goal to support such learning engineers, the evaluation presented in this article was still technical, by considering the qualities of the resulting agent, and not the ease of use of the workflow.

Finally, we highlight again that our workflow generates a "conversational module," which inherently does not constitute a complete educational system. Instead, it serves as a component within a broader educational framework. This module could take various forms: it might serve as a specific interactive exercise within a lesson led by a human teacher, be integrated into a tutorial agent that covers multiple subjects and different types of conversations, or it could be part of a traditional web-based interactive learning system with some conversational interface exercises, such as ours included. Overall, we argue that educational technology research should be more concerned with how to produce content for adaptive technology: real classrooms—at whatever level of education—need an enormous amount of content. On the other hand, technology-enhanced learning literature is full of very promising, very well-engineered systems that cover a tiny piece of the curriculum. The present work contributes specifically by showing how transformer-based language models can support the process of learning engineering for adaptive educational technology. Naturally, this does not address or remove all challenges in learning engineering, and further studies of such workflows will be interesting.

We hope that this work does spark interesting future work that builds on existing technological advances, and solid learning sciences foundation.

ACKNOWLEDGMENT

This article was produced by the IEEE Publication Technology Group. They are in Piscataway, NJ, USA.

REFERENCES

[1] C. Von Aufschnaiter, S. Erduran, J. Osborne, and S. Simon, "Arguing to learn and learning to argue: Case studies of how students' argumentation relates to their scientific knowledge," *J. Res. Sci. Teach.: Official J. Nat. Assoc. Res. Sci. Teach.*, vol. 45, no. 1, pp. 101–131, 2008.

[2] S. Erduran, S. Simon, and J. Osborne, "Tapping into argumentation: Developments in the application of Toulmin's argument pattern for studying science discourse," *Sci. Educ.*, vol. 88, no. 6, pp. 915–933, 2004.

[3] C. Chin, "Teacher questioning in science classrooms: Approaches that stimulate productive thinking," *J. Res. Sci. Teach.*, vol. 44, no. 6, pp. 815–843, 2007.

[4] P. Black and C. Harrison, "Feedback in questioning and marking: The science teacher's role in formative assessment," *Sch. Sci. Rev.*, vol. 82, no. 301, pp. 55–61, 2001.

[5] S. E. Toulmin, *The Uses of Argument*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[6] B. Mirzababaei and V. Pammer-Schindler, "Developing a conversational agent's capability to identify structural wrongness in arguments based on Toulmin's model of arguments," *Front. AI*, vol. 4, 2021, Art. no. 645516.

[7] I. Habernal and I. Gurevych, "Argumentation mining in user-generated web discourse," *Comput. Linguistics*, vol. 43, no. 1, pp. 125–179, 2017.

[8] R. Whately, *Elements of Logic*. London, U.K.: Longman, Green, Longman, Roberts and Green, 1897.

[9] B. Mirzababaei and V. Pammer-Schindler, "Learning to give a complete argument with a conversational agent: An experimental study in two domains of argumentation," in *Proc. Eur. Conf. Technol. Enhanced Learn.*, 2022, pp. 215–228.

[10] T. Wambsganss, T. Kueng, M. Soellner, and J. M. Leimeister, "ArgueTutor: An adaptive dialog-based learning system for argumentation skills," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2021, pp. 1–13.

[11] W. Wang, D. Arya, N. Novielli, J. Cheng, and J. L. Guo, "ArguLens: Anatomy of community opinions on usability issues using argumentation models," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2020, vol. 20, pp. 1–14.

[12] Z. H. Amur, Y. Kwang Hooi, H. Bhanbhro, K. Dahri, and G. M. Soomro, "Short-text semantic similarity (STSS): Techniques, challenges and future perspectives," *Appl. Sci.*, vol. 13, no. 6, 2023, Art. no. 3911.

[13] A. C. Graesser, K. VanLehn, C. P. Rosé, P. W. Jordan, and D. Harter, "Intelligent tutoring systems with conversational dialogue," *AI Mag.*, vol. 22, no. 4, pp. 39–39, 2001.

[14] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2017, vol. 30, pp. 6000–6010.

[15] S. Zhang, R. Fan, Y. Liu, S. Chen, Q. Liu, and W. Zeng, "Applications of transformer-based language models in bioinformatics: A survey," *Bioinf. Adv.*, vol. 3, no. 1, 2023, Art. no. vbad001.

[16] B. Mirzababaei and V. Pammer-Schindler, "An educational conversational agent for GDPR," in *Proc. 17th Educ. New Future: Mak. Sense Technol.-Enhanced Learn. Adoption*, 2022, pp. 470–476.
[17] S. Demetriadis et al., "Conversational agents as group-teacher interaction mediators in MOOCs," in *Proc. Conf. Learn. MOOCS*, 2018, pp. 43–46.
[18] F. Weber, T. Wambsganss, D. Rüttimann, and M. Söllner, "Pedagogical agents for interactive learning: A taxonomy of conversational agents in education," in *Proc. 42nd Int. Conf. Inf. Syst.*, 2021, pp. 1–17.
[19] D. B. Clark, V. Sampson, A. Weinberger, and G. Erkens, "Analytic frameworks for assessing dialogic argumentation in online learning environments," *Educ. Psychol. Rev.*, vol. 19, pp. 343–374, 2007.
[20] F. Loll and N. Pinkwart, "Guiding the process of argumentation: The effects of ontology and collaboration," in *Proc. Conf. Connecting Comput.-Supported Collaborative Learn. Policy Pract.*, vol. 1, 2011, pp. 296–303.
[21] T. Afrin, O. Kashefi, C. Olshefski, D. Litman, R. Hwa, and A. Godley, "Effective interfaces for student-driven revision sessions for argumentative writing," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2021, pp. 1–13.
[22] R. Flesch, "Marks of readable style: A study in adult education," *Teachers College Contributions Edu.*, vol. 897, 1943, Art. no. 69.
[23] O. Scheuer and B. M. McLaren, "CASE: A configurable argumentation support engine," *IEEE Trans. Learn. Technol.*, vol. 6, no. 2, pp. 144–157, Apr.–Jun. 2013.
[24] J. Lawrence and C. Reed, "Argument mining: A survey," *Comput. Linguistics*, vol. 45, no. 4, pp. 765–818, 2020.
[25] P. Poudyal, T. Goncalves, and P. Quaresma, "Experiments on identification of argumentative sentences," in *Proc. Int. Conf. Softw., Knowl. Inf., Ind. Manage. Appl.*, 2016, pp. 398–403.
[26] C. Stab and I. Gurevych, "Identifying argumentative discourse structures in persuasive essays," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 46–56.
[27] R. M. Palau and M. F. Moens, "Argumentation mining: The detection, classification and structure of arguments in text," in *Proc. 12th Int. Conf. AI Law*, 2009, pp. 98–107.
[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter ACL: Hum. Lang. Technol.*, 2019, vol. 1, pp. 4171–4186.
[29] T. Wambsganss, A. Janson, T. Käser, and J. M. Leimeister, "Improving students argumentation learning with adaptive self-evaluation nudging," *Proc. ACM Hum.- Comput. Interact.*, vol. 6, no. CSCW, 2022, Art. no. 520.
[30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[31] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2013, vol. 26, pp. 3111–3119.
[32] T. Wambsganss, C. Niklaus, M. Söllner, S. Handschuh, and J. M. Leimeister, "A corpus for argumentative writing support in German," in *Proc. 28th Int. Conf. Comput. Linguist.*, 2020, pp. 856–869.
[33] M. Xia, Q. Zhu, X. Wang, F. Nie, H. Qu, and X. Ma, "Persua: A visual interactive system to enhance the persuasiveness of arguments in online discussion," *Proc. ACM Hum.- Comput. Interact.*, vol. 6, no. CSCW 2, 2022, Art. no. 319.
[34] W. A. Abro, A. Aicher, N. Rach, S. Ultes, W. Minker, and G. Qi, "Natural language understanding for argumentative dialogue systems in the opinion building domain," *Knowl. Based Syst.*, vol. 242, 2022, Art. no. 108318.
[35] M. Lippi and P. Torroni, "Argumentation mining: State of the art and emerging trends," *ACM Trans. Internet Technol.*, vol. 16, no. 2, pp. 1–25, 2016.
[36] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2019, pp. 3982–3992.
[37] N. Reimers and I. Gurevych, "Making monolingual sentence embeddings multilingual using knowledge distillation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 4512–4525.
[38] F. Loll, N. Pinkwart, O. Scheuer, and B. M. McLaren, "How tough should it be? Simplifying the development of argumentation systems using a configurable platform," in *Educational Technologies for Teaching Argumentation Skills*. Sharjah, United Arab Emirates: Bentham Sci. Pub., 2012, pp. 169–197.
[39] C. Schulz, S. Eger, J. Daxenberger, T. Kahse, and I. Gurevych, "Multi-task learning for argumentation mining in low-resource settings," in *Proc. Conf. North Amer. Chapter ACL: Hum. Lang. Technol.*, 2018, vol. 2, pp. 35–41.
[40] A. Lauscher, G. Glavaš, and S. P. Ponzetto, "An argument-annotated corpus of scientific publications," in *Proc. 5th Workshop Argument Mining*, 2018, pp. 40–46.
[41] M. Specht, M. Kravcik, L. Pesin, and R. Klemke, "Authoring adaptive educational hypermedia in WINDS," in *Proc. Workshop Adaptivität und Benutzermodellierung Interaktiven Softwaresystemen*, 2001, pp. 1–8.
[42] P. Brusilovsky, S. A. Sosnovsky, M. Yudelson, and G. Chavan, "Interactive authoring support for adaptive educational systems," in *Proc. Conf. AI Educ.*, 2005, pp. 96–103.
[43] A. Mannekote, M. Celepkolu, J. B. Wiggins, and K. E. Boyer, "Exploring usability issues in instruction-based and schema-based authoring of task-oriented dialogue agents," in *Proc. 5th Int. Conf. Conversational User Interfaces*, 2023, pp. 1–6.
[44] T. A. Van Dijk, "Text and context: Explorations in the semantics and pragmatics of discourse," London, U.K.: Longman, 1977.
[45] I. Wolfbauer, V. Pammer-Schindler, K. Maitz, and C. P. Rosé, "A script for conversational reflection guidance: A field study on developing reflection competence with apprentices," *IEEE Trans. Learn. Technol.*, vol. 15, no. 5, pp. 554–566, Oct. 2022.
[46] D. H. Jonassen and B. Kim, "Arguing to learn and learning to argue: Design justifications and guidelines," *Educ. Technol. Res. Develop.*, vol. 58, pp. 439–457, 2010.
[47] D. Long and B. Magerko, "What is AI literacy? Competencies and design considerations," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2020, pp. 1–16.
[48] S. J. Russell, *Artificial Intelligence a Modern Approach*. London, U.K.: Pearson, 2010.
[49] D. Adamson, G. Dyke, H. Jang, and C. P. Rosé, "Towards an agile approach to adapting dynamic collaboration support to student needs," *Int. J. AI Educ.*, vol. 24, no. 1, pp. 92–124, 2014.
[50] R. Kumar, J. Beuth, and C. Rosé, "Conversational strategies that support idea generation productivity in groups," in *Proc. Conf. Connecting Comput.-Supported Collaborative Learn. Policy Pract.*, 2011, vol. 1, pp. 398–405.
[51] C. Stab and I. Gurevych, "Parsing argumentation structures in persuasive essays," *Comput. Linguistics*, vol. 43, no. 3, pp. 619–659, 2017.
[52] M.-F. Moens, E. Boiy, R. M. Palau, and C. Reed, "Automatic detection of arguments in legal texts," in *Proc. 11th Int. Conf. AI Law*, 2007, pp. 225–230.
[53] T. Wambsganss, C. Niklaus, M. Cetto, M. Söllner, S. Handschuh, and J. M. Leimeister, "AI: An adaptive learning support system for argumentation skills," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2020, pp. 1–14.

Behzad Mirzababaei received the M.A. degree in computer science from the University of Tehran, Tehran, Iran, in 2014. He is currently working toward the Ph.D. degree in computer science with the Graz University of Technology, Graz, Austria.

He is currently a Researcher with the Know-Center, a not-for-profit research company. His research interests include technology-enhanced learning, argument mining, tutoring systems, and educational chatbots.

Viktoria Pammer-Schindler received the doctoral and Habilitation degrees in computer science from the Graz University of Technology, Graz, Austria, in 2010 and 2019, respectively.

Since 2012, she has been a Research Area Head at the Know-Center, a not-for-profit research company. Since 2020, she has been an Associate Professor at TU Graz, Graz. Her research interests include socio-technical interventions for learning and knowledge work, and the intersection of technology-enhanced learning and human–computer interaction.

Dr. Pammer-Schindler was the President of the International Alliance to Advance Learning in the Digital Era, a Member of the European Association of Technology-Enhanced Learning's Managing Committee, and a Subcommittee Chair with the ACM Conference on human factors in computing.