IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 September 2024, accepted 8 October 2024, date of publication 14 October 2024, date of current version 24 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3479883

RESEARCH ARTICLE

An Educational Dialogue System for Visually Impaired People

PIER FELICE BALESTRUCCI[1], ELISA DI NUOVO[2], MANUELA SANGUINETTI[3], LUCA ANSELMA[1], CRISTIAN BERNAREGGI[4], AND ALESSANDRO MAZZEI[1]

[1]Dipartimento di Informatica, Università degli Studi di Torino, 10149 Turin, Italy
[2]Joint Research Centre, European Commission, 21027 Ispra, Italy
[3]Dipartimento di Matematica e Informatica, Università degli Studi di Cagliari, 09124 Cagliari, Italy
[4]Laboratorio Polin, Dipartimento di Matematica, Università degli Studi di Torino, 10123 Turin, Italy

Corresponding author: Pier Felice Balestrucci (pierfelice.balestrucci@unito.it)

This work has been developed for the project NoVAGraphS (Non-Visual Access to Graphical Structure), funded by the grant Progetto CRT 2021.1930 from Fondazione CRT (Cassa di Risparmio di Torino).

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the University of Turin - Comitato di Bioetica d'Ateneo under Application No. 0576623.

ABSTRACT Finite State Automata (FSA) serve as a fundamental mathematical model of computation, commonly explored in computer science degree programs. Traditionally, FSA are represented through state diagrams or tables, which, in educational settings necessitating assistive technologies, pose accessibility challenges, particularly for Visually Impaired People (VIP). This work is part of a broader initiative focused on creating tools for inclusive access to scientific information. Its main goal is to explore the effectiveness of a dialogue system (DS) as an alternative method for conveying FSA information to VIP, aiming to enhance their comprehension and user experience. To achieve this, a rule-based DS tailored for facilitating FSA access, with a primary focus on VIP as end users, was developed. The research involved an A/B test comparing participants' comprehension of FSA using the rule-based DS versus standard tabular representations. Statistical analysis was also carried out to evaluate the performance differences between the two methods. The findings indicate that communication through the DS significantly outperforms the tabular representation, establishing it as a viable and effective alternative for both VIP and non-VIP. Although VIP participants displayed slightly varying performance depending on the questions, their feedback favoured the DS for its ease of use and overall user satisfaction. Additionally, the study analysed the capabilities and limitations of popular Large Language Models (LLMs) in describing graphical structures like FSA. Despite their general effectiveness in language tasks, LLMs proved inadequate for accurately and consistently describing FSA, highlighting the need for more controlled and explainable DS approaches in educational contexts.

INDEX TERMS Assistive technologies, chatbots, educational technology, natural language processing.

I. INTRODUCTION

In educational settings, ensuring equal opportunities and access to learning materials for all students is crucial. However, accessibility challenges, particularly for Visually Impaired People (VIP[1]), arise, especially in understanding *graphical structures*, such as diagrams with internal structures (e.g., tables, trees, UML, E-R, circuits). Image descriptions or alt-texts, commonly used for accessibility, may prove unsatisfactory. Indeed, a static textual description, such as "alt text", is a general description that can be

The associate editor coordinating the review of this manuscript and approving it for publication was Arianna DUlizia.

[1]See Section VIII for a list of the acronyms used in the paper.

extremely long since it has to describe in text all relevant graphic features of the image. Such a description could be hard to "navigate" to extract information and usually is complementary to the image, and not thought of as a substitute for the image.

While tactile and haptic devices are recognised as limited [1], Natural Language Processing and Generation (NLP/G) appear promising in effectively delivering graphical information. Moreover, the availability of speech-to-text and text-to-speech facilities in electronic devices suggests their potential in reducing accessibility barriers when combined with NLP/G technologies. In this context, Dialogue Systems (DSs) play a crucial role in education by employing various techniques to support students in diverse learning environments [2]. They leverage technology to enhance accessibility to educational content, particularly benefiting students who prefer independent study. Additionally, DSs facilitate interactions for students with special needs, providing support in guided reading, learning disabilities, and dyslexia [3], [4]. In Computer Science classes, a significant number of students face accessibility challenges, prompting recent legislation aimed at ensuring accessibility in information technology tools.[2]

This article addresses the accessibility of Finite State Automata (FSA), which are a mathematical model of computation typically taught in Computer Science degrees[3] and are paradigmatic for other graphical structures.[4]

FSA are usually formalised as a quintuple: (1) a finite set of states, Q, (2) a finite set of input symbols, Σ, (3) a transition function, $\delta : Q \times \Sigma \to Q$, which maps each state and input symbol to a new state, (4) a start state, $q_0 \in Q$, (5) a set of accepting states (also called final states), $F \subseteq Q$.

FSA can be represented as a diagram or a table representing the connections (i.e., transitions) between each state. In educational books, e.g., Hopcroft et al. [5], FSA are typically graphically rendered via state diagrams. As also mentioned above, however, this type of structure may be difficult to access for VIP. Indeed, VIP face several challenges because they must understand FSA solely based on descriptive textual information provided next to the graphical representation, which is often fragmented. One of the most agile and accessible ways for VIP to access a graphical representation is by using the State Table (ST) in HTML. In fact, VIP can then use specific speech synthesizers, called *screen readers*, to listen to the content of the cells of the HTML tables and use the keyboard to navigate between the cells. However, this type of exploration can be time-consuming and exhausting for VIP. On the contrary, having a specific textual representation that allows the exploration of graphs can not only be faster, but also preferable (see Section V-D). In addition, it can also cater to various other types of disabilities, such as motor disabilities. However, little to no attention has been drawn so far to the use of dialogue-based applications for accessing FSA and, more broadly, for accessing information in graphical structures. Modern Large Language Model (LLM)-based systems do not address this gap, and their applicability, especially in terms of robustness and trustworthiness, remains to be demonstrated.

For the reasons outlined above, this article proposes a new methodology to access FSA: it investigates the extent to which a rule-based DS can effectively describe FSA, enabling a *dialogic* exploration that can focus on elements that may be challenging to comprehend through more traditional representations. This rule-based DS is first presented to describe FSA, making it accessible to VIP by offering detailed and consistent descriptions. To ensure accessibility, the DS was designed in collaboration with a visually impaired expert. A comprehensive experimental study was conducted to test the hypothesis regarding the enhanced accessibility of FSA through the DS, involving both VIP and non-VIP participants and comparing DS interactions with tabular exploration via the ST. For the experiment, a web-based accessible textual DS was devised, accessible to VIP using standard speech technologies, typically a screen reader for listening and a keyboard for typing. This approach allows to evaluate the impact of dialogic interaction without altering the VIP's final physical speech interaction.[5] Furthermore, an additional study that illustrates how the use of rule-based DSs might be more suitable than the current Large Language Models for describing FSA in an educational context is presented.

The research questions driving this study are thus outlined below:

1) Is interacting with a DS more effective than using a ST for acquiring information about FSA?
2) Which method of exploration (DS or ST) is preferable for VIP to acquire information about FSA?
3) Can recent technologies such as LLMs be leveraged to address accessibility issues for VIP in an educational context?

To our knowledge these three research questions have not been addressed in previous work and their answers can open up new research lines.

In line with the research objectives, Section II first reports the state of the art concerning the specific problem of accessing graphical structures by VIP and the assessment of LLMs in the educational field. Following this, Section III describes the development of the textual rule-based DS with a particular focus on the design choices and the key features of the web interface used for the interactions. Sections IV and V provide a detailed account of the user study carried out

[2]See, as an example, the European Union directive on the accessibility of the websites and mobile applications of public sector bodies: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016L2102

[3]FSA is a fundamental topic for STEM students since it is contained in the standard syllabus for computer science curriculum provided by ACM https://dl.acm.org/doi/pdf/10.1145/3664191

[4]This research has been developed as part of the NoVAGraphS (Non Visual Access to Graphical Structures) project: http://www.integr-abile.unito.it/en/progetto-novagraphs/

[5]The use of text-to-speech technology which is not specifically designed for VIP can degrade their user experience, as also observed in previous studies [6], [7].

to test the DS, while Section VI complements the analysis showing the main limitations of current LLMs in describing FSA. A discussion of the main findings and limitations of this study finally completes this work (Section VII).

II. RELATED WORK

This section begins with a discussion of accessible solutions for communicating graphical information to VIP. Following this, an overview of recent studies involving LLMs in the educational field is provided.

A. ACCESSIBLE SOLUTIONS FOR VIP

Haptic representations have been proposed in multimodal systems for communicating graphical information in motion [8], exploring graphs [9], chemical formulas [10], function graphs [11], geometric shapes [12] and various types of diagrams [13]. Sonification has been studied to explore generic images [14], function graphs [1], elementary geometric shapes [15], maps [16] and various types of diagrams [17]. Textual descriptions of images have been studied in particular for representing generic images [18], chemical formulas [19], electronic circuits [20], function graphs [21] and statistical diagrams [22]. These studies point out two main limitations of such solutions: a thorough knowledge of the domain and characteristics of non-visual perception is required on the part of the person making the textual description. Second, the cognitive load to understand the textual description of a complex image is very high since the exploration is sequential.

A number of academic studies and commercial software applications applied NLG to describe to VIP some specific structures, typical of scientific communication. For instance, different NLG techniques have been applied to produce descriptions of bar charts [23]. Other studies consider the problem of communicating mathematical expressions by using mathematical sentences, that are natural language sentences verbalising the expressions [6].

To the best of the authors' knowledge, there is only one existing study on the application of a DS to access FSA [24]. However, that preliminary investigation focused on the task of Dialogue Acts annotation and the evaluation of the resulting dialogue corpus. There appears to be no other research on the use of DSs for describing the information contained in graphical structures.

B. ASSESSING LLMs IN THE EDUCATIONAL FIELD

Several recent studies have been conducted about the use of LLMs in the educational field after the release of ChatGPT [25] and its impressive performance in several domains such as programming [26], English comprehension [27] and critical thinking [28]. Reference [29] provides an overview of ChatGPT's performance in various domains and tasks, while also highlighting the potential risks that such a tool may pose to students and educators. Reference [30] assesses that ChatGPT can provide assistance as a virtual intelligent system by flanking educators and offering personalised support, despite they also raise concerns on the serious issues of lack of reliability and hallucinations. In [31] the emphasis is also placed on the role of ChatGPT in empowering learners with disabilities such as VIP through the integration of this tool with speech-to-text and text-to-speech technologies. However [31] do not fail to underline the difficult challenges in distinguishing misleading but convincingly-presented information from sensible content.

Section VI explores the possibility to use LLMs to explain a FSA through dialogic interactions.

III. A RULE-BASED DIALOGUE SYSTEM

When it comes to building a task-oriented DS, there are various alternatives such as leveraging rule-based systems or frames. More recently, LLMs have become a common choice in the development of agents capable of handling specific tasks through dialogue. The main objective of this work is to investigate whether a dialogic representation can be better than a tabular one, especially in terms of accessibility to VIP. For this reason, the focus is mainly on the design of a rule-based DS, using the Artificial Intelligence Markup Language (AIML) [32], a declarative language based on the pattern-matching paradigm. It is well known, in fact, that rule-based systems offer by design consistent and controlled responses, at the expense of flexibility and scalability. Nonetheless, systems of this kind would readily allow us to investigate the initial hypothesis of determining whether a dialogue-based interaction is indeed more effective than a tabular representation for VIP. This section describes the implementation of two specific FSA descriptions in AIML and the design of the *happy paths*—i.e., a number of prototypical dialogues in which the DS is able to provide all the requested information.[6]

To ensure precise and specialised descriptions of FSA concepts, terminology, definitions, and procedures were based on an influential textbook on formal languages that offers a comprehensive introduction to automata theory [5].

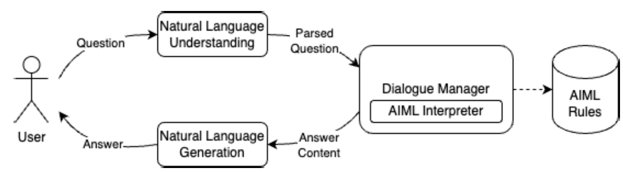

FIGURE 1. Dialogue system architecture.

The use of the DS involves several key steps, as illustrated in Fig. 1. First, the user submits a question, which undergoes pre-processing to remove special characters and white spaces. The system then employs a Natural Language Understanding module that searches for patterns within the user's input to determine the intent and identify relevant entities. Following this, the AIML Interpreter searches for the corresponding

[6]http://xunitpatterns.com/happy%20path.html

AIML rule that matches the detected keywords in the pattern. The Dialogue Manager coordinates the conversation flow, selecting the appropriate response. Finally, the Natural Language Generation module delivers this predefined response to the user, and the DS sends it back as the final output.[7] The information regarding the FSA is encoded in AIML rules.

```
<category>
  <pattern>HOW MANY STATES *</pattern>
  <template>
    There are a total of 3 states.
    q0, q1, and q2. q0 is both initial and final state.
  </template>
</category>
```

FIGURE 2. AIML rule example.

As an example of a rule, Fig. 2 reports the one that was applied more frequently in the experimentation (see Section V). This rule triggers the answer in the template tag to various questions such as *How many states are there?* or *How many states does the automaton have?*.[8] This rule catches all the questions starting with *How many states* and followed by one or more words (* is a wild card and it can assume any words). When a pattern is triggered by the user input, the DS will answer with a suitable template. Note that AIML, that is essentially based on a regular expression mechanism, also provides a number of facilities through the use of specific tags, as SRAI, CONDITION and RANDOM, that allow for an easier management of dialogue interactions. SRAI allows for a "call" to other AIML rules, improving system engineering, CONDITION selects the answer to a specific pattern considering the value of some local variables, and RANDOM adds non-deterministic rules improving system naturalness.

A first set of rules has been created by several happy paths designed to cover different task-related topics. A happy path refers to the expected path of a conversation, where everything goes according to plan without encountering any error or exception. All the happy paths can be grouped into three macro-categories: (1) *greetings*—i.e., introduction and greetings to users, (2) *how to*—i.e., questions explaining how to use the DS, (3) *description*—i.e., all the questions about the automaton, e.g., description, accepted language, states. This initial set has been expanded through an alpha test in which three in-house domain experts with a PhD degree in Computer Science volunteered to interact with the DS. One of the alpha tester is completely blind. As a result of this in-house test, the DS is actually composed of 103 rules. Some examples of dialogue just discussed can be found in Table 1 divided by category. See Table 22 in Appendix for more examples of AIML rules.

[7]The DS engine was developed by using Node.js and the open source library aiml-high https://github.com/gleuch/aiml-high to interpret the AIML rules. The DS can be freely downloaded at https://github.com/Reasoning-NLG-Unito/NoVAGraphs_

[8]Please note that the questions are reported verbatim as they are actual questions asked by the participants.

A. SYSTEM INTERFACE

When designing the web interface to encapsulate the DS, some important considerations were taken into account, in order to meet the needs of VIP. First of all, most VIP users have very good skills in using a screen reader, which is a specific software designed for vocal navigation of textual interfaces. In fact, VIP generally use a screen reader for accessing information and a keyboard for providing inputs. A modular approach was thus preferred, concentrating on the design and implementation of a textual DS while allowing VIP to use their standard input-output interface. Additionally, the developed web interface fully adheres to Web Content Accessibility Guidelines 2.1,[9] thus fully accessible to VIP. Fig. 3 shows the DS interface. The input form for writing text and the submission button were designed to ensure easy accessibility for VIP as they navigate through the entire page. The DS answer is conveniently located just below for quick reference. Beneath the response, a table, navigable through HTML tags, displays the dialogue history, including the input number, the input itself, and the corresponding response.

IV. EXPERIMENTAL PROTOCOL

The aim of this experimental phase is twofold: demonstrating the DS effectiveness over a graphical representation and evaluating the coverage and efficacy of the DS rules.

An A/B test dividing the participants in two groups was set up. Each participant explored two different automata with similar complexity: Automaton 1 has 5 states and 5 transitions, while Automaton 2 has 3 states and 3 transitions. The automata have the same alphabet made of 0s and 1s. The two automata are comparable considering that Automaton 2 has more difficult (less symmetrical) transitions despite having less transitions and states than Automaton 1. Participants in both groups explored the two automata just introduced. The difference, as explained more in detail below, is only based on which method they used to explore Automaton 1 and Automaton 2, whether it was the DS or the ST. The ST contains all relevant information, including the states, initial and final states, and transitions between them. This information is presented into an HTML table, which provides a viable way for VIP to access it by exploring its headers and columns. Additionally, the ST representation included also a description of the language accepted by the automaton.

For the experimentation, 33 volunteer participants (26 non-VIP and 7 VIP) were recruited. The requirements to participate, as stated in the invitation letter (see Table 15 in Appendix), were fluency in English and knowledge of FSA.[10] Out of the 33 participants, 65.6% fall within the age range of 25–34, 6.3% are 45–54 years old, 12.5% 18–24 years old and 15.6% 35–44 years old.

To carry out the A/B test, the two groups of participants, ABC and XYZ, were pointed to two websites: ABC and XYZ.

[9]https://www.w3.org/WAI/standards-guidelines/wcag/

[10]Studies concerning assistive technologies for VIP have a total average number of participants of 22.3 and they often are sighted persons (30% of the studies analysed in [33]).

TABLE 1. Examples of user input and system output.

Category	User Input	System Output
greetings	Hi!	Hi. I am *NoVABot*. I'm programmed to answer questions to describe graphical structures, and in particular, automata. Ask me to describe you in general Automaton 1 or a more specific question about it.
howto	How can you help me?	Examples of questions may be: Briefly describe the automaton, How many states are there?, Is there an arc from q0 to q1?, How is the automaton represented? For more information, click on Homepage tab.
description	What is the accepted language?	The language accepted by this automaton is made of zero or more words formed by a sequence of a triple of 1s followed by a pair of 0s.
description	How many states are there?	There are a total of 5 states. q0, q1, q2, q3, and q4. q0 is both the initial and final state.

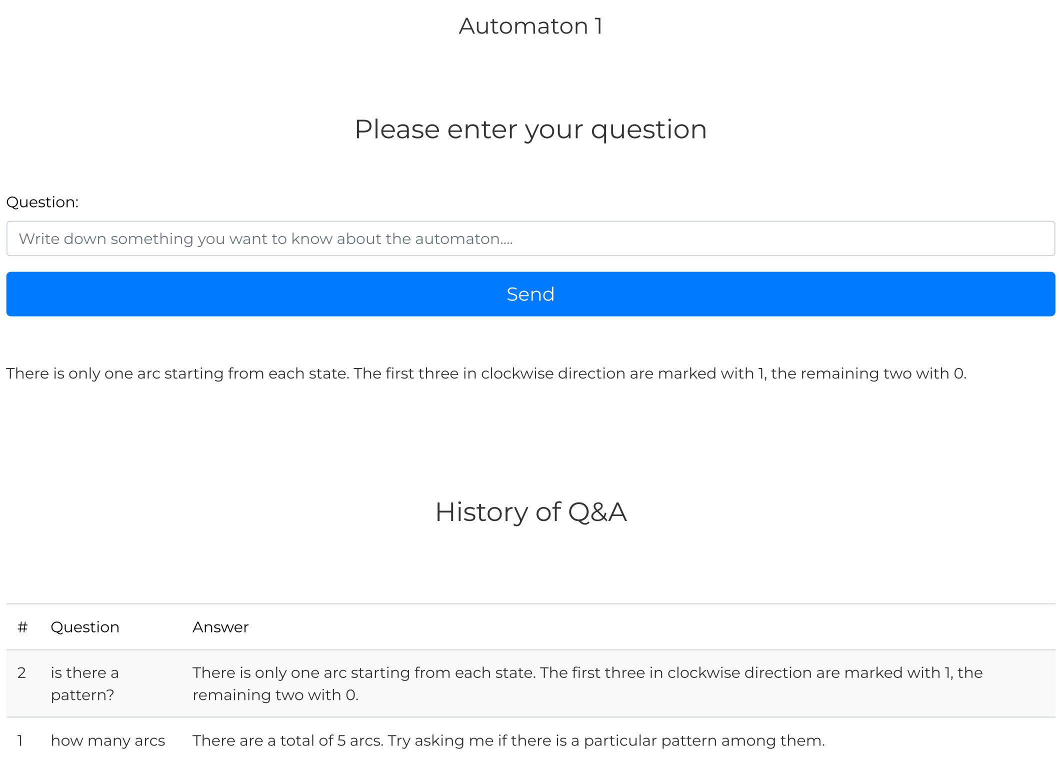

FIGURE 3. System interface for the DS representing Automaton 1. On top there is a text box that users can use to write down their questions. On bottom an accessible table to explore the previous interactions.

Both ABC and XYZ websites consist of three pages: one page containing the explanation of the experiment and providing a link to a Google Form questionnaire,[11] and two more pages with the two automata—one to be explored via the ST, and the other via the DS. The DS is encapsulated in the web interface described in Section III-A, while the ST is displayed as a navigable HTML table (see Fig. 4). Input symbols from the alphabet are used as table headers. Each row corresponds to a state in the FSA. The adjacent column for each row indicates the next state linked to it, or "/" if there is no link.

Participants could complete the assessment questions using both the ST and the DS conversation history, as the goal was not to test their memory, but their ability to acquire knowledge on FSA using these two methods. The websites ABC and XYZ differ by the order of the pages containing the automata and their association to ST and DS. In particular, in website ABC, the DS page appears first and enables the exploration of Automaton 1, while the second page proposes the exploration of Automaton 2 via the ST. In website XYZ, the order is reversed, thus having the ST in the first page to

[11] ABC group questionnaire: https://forms.gle/Psk1BYdUbqVZL9Ec9. XYZ group questionnaire: https://forms.gle/gMuriMwBajcXygXdA

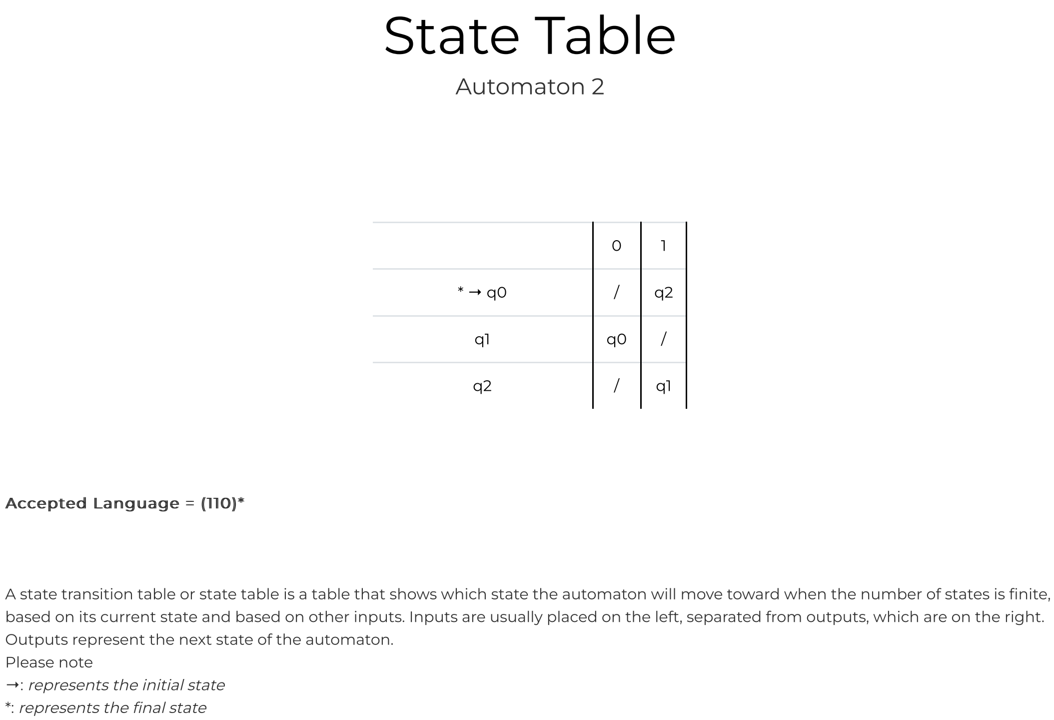

FIGURE 4. System interface for the ST representing Automaton 2. A Table where the label transitions are on columns and the states on the rows. $*$ and $\rightarrow$ indicate the final and initial state, respectively.

explore Automaton 1 and the DS in the second page to explore Automaton 2.

Google Forms was employed to anonymously gather responses to a questionnaire intended to assess user understanding of the two automata, due to its accessible interface for VIP.

More precisely, the questionnaire comprised 10 closed-ended questions per automaton:

- Five multiple-choice questions, referred to as *knowledge questions*, aimed at testing the participants' understanding of the automaton (e.g., *what is the initial state?*).
- Five questions, referred to as *reasoning questions*, aimed at testing participants' reasoning abilities about the automaton. They comprised four true/false questions and one multiple-choice question (e.g., *if the initial state is $q1$ instead of $q0$, is the accepted language the same?*).

In addition, three open-ended questions were included:

- One question asking for a description of the automata; and
- Two questions for collecting feedback on the interaction with the DS.

Finally, the form included some profiling questions (e.g., age, prior knowledge of FSA). All the questions can be found in Appendix VIII-A.

To ensure the quality of the answers, the participants who posed less than half of pertinent questions, and who covered less than half of the key concepts of the (quintuple of the) FSA were excluded. As a result, two participants in group XYZ were excluded. Ultimately, the ABC group comprised 16 participants, while the XYZ group included 15 participants.

V. RESULTS

This section analyzes the questionnaire results. Initially, a quantitative analysis of the closed-ended questions is conducted, followed by a thematic analysis [34] to the open-ended question related to the description of the automaton. Then, qualitative analysis is performed on the received feedback and interactions with the DS, with a particular focus on the results obtained from VIP participants.

A. QUANTITATIVE ANALYSIS

For the closed-ended questions, a score of 1 was assigned for correct and complete answers, while 0 points were given for incorrect or partially incorrect responses. Being the closed-ended questions 10 in number for each automaton, the maximum score obtainable is 10. Participants obtained high scores overall.

TABLE 2. Results obtained by participants using ST and DS with the closed-ended questions – Average (standard deviation).

	Knowledge	Reasoning	All
ST	4.65 (0.95)	4.06 (0.96)	8.71 (1.49)
DS	**4.77** (0.43)	**4.16**(0.90)	**8.94** (1.06)

As it can be seen in Table 2, the scores are similar between DS and ST and above 8 out of 10; in particular, the scores obtained by participants when using the DS are consistently higher than when using the ST, both considering the whole set of questions and the sets of knowledge and reasoning questions alone. On the one hand, these high scores confirm that the users, regardless of the specific automaton and method of exploration, were actually knowledgeable about automata. On the other hand, the DS proves to be competitive over a more traditional ST, even considering both VIP and non-VIP. The significance of such difference was assessed with a two-tail non-parametric Wilcoxon signed-rank test [35], as a Kolmogorov-Smirnov normality test reports that the values do not follow a normal distribution (e.g., $D^{DS}_{knowledge} = 0.78$, $p < .01$). The results indicate that participants answered significantly better when exploring the FSA via the DS than via the ST both on knowledge and reasoning questions ($W_{knowledge} = 10$, $W_{reasoning} = 96$, $p < .005$).

The answers to open-ended questions on automata description were also analyzed quantitatively. This was achieved through a hybrid deductive/inductive thematic analysis approach [34], [43] reaching five themes, which describe five key concepts of automata: accepted language, states, transitions, alphabet, and graphical representation. All provided descriptions were analyzed to determine the presence and accuracy of key concepts. Each description was scored with 1 point for a correct and complete explanation of a key concept, and 0 points for missing or incorrect information. Thus, a score of 5 corresponds to a full description of the automaton. For example, the user description *"It is an automaton that accepts the sequence* 110 *repeated an arbitrary number of time* (110)∗*."*, containing correct information only on the accepted language but not mentioning states, transitions, alphabet and graphical representation, scores 1 out of 5.

The average scores are shown in Table 3. As it can be noted, both the mean scores for DS and ST are quite low, slightly less than 2 out of 5. The low scores may be attributed to the lack of explanation provided to participants regarding how the comprehensiveness of their responses, particularly in terms of covering the five key concepts, would be evaluated.Nonetheless, in these open-ended questions as well, the DS scores are better than the ST ones. The observed difference may indicate that individuals who interact with a DS are more likely to give a more complete description. A two-tail non-parametric Wilcoxon signed-rank test indicates that descriptions resulting from the interactions with the DS are significantly better than using the ST ($W = 45.5$, $p < .005$).

B. QUALITATIVE ANALYSIS

This subsection analyses the feedback received from the final two open-ended questions administered to participants. They were asked:

TABLE 3. Results obtained by participants using ST and DS with the open-ended questions about automata description – Average (standard deviation).

	Automata description
ST	1.74 (1.20)
DS	**1.83** (1.04)

- One question prompting for information about non-answered questions.
- One question on how they would improve the interaction.

20 out of 31 participants gave feedback on the interaction with the DS. Of these, 5 (25%) explicitly stated that the interaction was good, complete or satisfactory. Among the DS features, users most appreciated its ability to recommend follow-up questions or prompting possible questions when the user's question was not clear enough. The feature that 6 users (30%) deemed useful to improve the interaction was the ability to deal with a larger coverage of synonyms and formulations. In addition, 3 users (15%) felt that presenting in the DS page a most-asked-question snippet would improve the interaction, as it would provide them with examples of questions. Other 3 users felt than the answers were sometimes too verbose or not tailored enough to the question/request. As far as VIP are concerned, 4 out of 7 expressed their wish to being able to use a larger variety of synonyms and periphrases, 1 left no feedback, 1 would have improved the interaction by providing more examples of accepted questions and 1 said that the interaction was satisfactory.

Looking at the questions which were not satisfactorily answered by the DS, as noticed by the 6 participants above, the majority were not pertinently answered because the DS was not able to handle all the synonyms (e.g., it recognised *transitions* and *arcs* as synonym, but not *links*) and all possible ways to address the same question (e.g., *What it its optimal spacial representation?*, *Describe graphically the automaton.*, *How are the states positioned?*, all expecting an answer like *This automaton can be represented as a triangle, with states as vertices.*). This was mainly due to the limitations intrinsic to the system's architecture (a rule-based DS).

As commented by three participants, sometimes the DS answers are not perfectly tailored to specific questions. This is due to the effort to create the minimal number of patterns that would cover the maximum number of questions. In addition, having an educational purpose, the DS is keen to add information instead of answering rigidly the question.

C. INTERACTION ANALYSIS

The collection of the interactions generated during the experimentation described above formed a corpus; such interactions are further analysed in this subsection.[12] The

[12] The resulting corpus of interactions, with additional annotations, can be freely requested through the following link: https://zenodo.org/records/10822733

collected corpus consists of 31 human-machine dialogues, for a total of 700 turns (i.e., 350 user questions/requests and 350 DS answers) consisting on average of 22.58 turns per dialogue.

TABLE 4. Basic corpus statistics.

		VIP	Non-VIP	All
Dialogues		7	24	31
Turns		214	486	700
Turns/Dialogue	Avg.	30.57	20.25	22.58
	Min.	20	8	
	Max.	56	34	
	σ	12.63	8.37	10.24
Tokens/User's turns	Avg.	4.77	6.14	5.83
	Min.	2.64	4.73	
	Max.	10.20	8.67	
	σ	2.51	1.26	1.68

Upon an initial examination of the gathered dialogues, it becomes evident that they predominantly consist of question-answer pairs. This pattern likely stems from the task setting, where users aim to extract information about the FSA by engaging with the DS. To add depth to the understanding of the dialogue dynamics, Table 4 reports some basic corpus statistics, highlighting quantitative differences between the two subgroups. VIP dialogues, as revealed by the statistics, exhibit longer overall but shorter individual turns. These variations in the number of turns per dialogue and the number of tokens per turn between the subgroups are statistically significant ($p = 0.046$ and $p = 0.007$, respectively) according to Mann-Whitney U test, accounting for two independent groups with non-normally distributed data and small sample sizes.

Analysing the questions that the DS was not able to answer, off-topic questions (e.g., *How many regions are there in Italy?*) or on-topic questions but in a language different from English (e.g., in Italian *Vorrei una descrizione di Automa1* meaning 'I would like a description of Automa1'), were found. These questions triggered as response *I don't know, try asking something like 'What is the initial state?'*.

Sometimes the NLU component misinterpreted the meaning giving an answer which unintentionally violated one or more of the Grice's conversational maxims [37], leading to a suboptimal communication.

In Table 5.A, the DS response violates the maxim of manner by failing to initiate the answer with a direct negation, resulting in ambiguity. In Table 5.B, the maxim of relation is compromised as the DS response provides no information about the transitions connecting q4 and q0, providing instead the transitions starting from q0. Lastly, in Example 5.C, the DS response violates the maxim of quantity, presenting an incomplete answer by omitting crucial information about transitions between states, which can hinder the user's understanding of the automaton behavior.

As expected, there were also questions not included in the designed *happy paths*, e.g., *is q0 linked to q0?*, which could be paraphrased as *are there any cycles in q0?*.

D. SUBGROUP ANALYSIS ON VIP

This subsection evaluates the responses and feedback from the subgroup consisting of the 7 VIP (4 used the ABC version and 3 the XYZ version). Recognising that the small sample size precludes statistically significant inferences, the results will be examined by discussing the data in relation to the findings presented in Sections V-A–V-B and the comments they have provided. To gain additional insights from this analysis, 3 further questions for the VIP alongside the questionnaire they had to complete as the other participants were added. Specifically, the additional questions are:

Q1. *Which of the two methods did you prefer the most considering the easiness to acquire information about the automata?*

Q2. *How would you rate the state table on a scale with 1 being "very poor" and 5 being "very good"?*

Q3. *How would you rate the dialogue system on a scale with 1 being "very poor" and 5 being "very good"?*

Of the 7 VIP, 57.1% fall within the age range of 35–44, with the remaining 14.3% each distributed between the age groups of 25–34, 45–54, and 18–24 (1 per age group). The entire population is male and non-native English speakers; however, all of them claim a good proficiency in English. Furthermore, 71.5% hold a BSc, 14.3% a PhD and 14.3% a high-school diploma, all in computer science.

The results in Tables 6, 7 and 8 show the average scores given by VIP per set of questions for the DS and ST. They suggest that VIP have slightly better comprehension and provide better descriptions of the automaton described through the DS in every case except for reasoning questions.

Nonetheless, while considering the answers to the three additional questions posed to VIP participants, it emerged that all of them preferred the DS as method to acquire information about the automata, due to its easiness of use (see question Q1.). Their preference overall was also confirmed by the average ratings assigned to ST and DS (questions Q2. and Q3.), i.e., 2.71 and 4.57, respectively, as also reported in Table 8.

VI. AN EVALUATION OF LLMs FOR DESCRIBING FSA

Previous sections presented the design and experimental evaluation of a rule-based DS conceived as an alternative tool to facilitate the comprehension of FSA. However, it is acknowledged that rule-based DSs have recently been surpassed by LLMs in many practical applications.

This section explores the capabilities and, crucially, the main limitations exhibited by LLMs in the specific task of describing FSA. This study aims to investigate whether LLMs can also be employed in this envisioned usage scenario and if they can be easily utilised to address specific questions related to FSA. The objective of this section is to evaluate how LLMs answer to queries about a FSA and to verify not only their accuracy, but also whether they can provide well-reasoned and coherent justifications of their responses (both aspects that are paramount in an educational context).

TABLE 5. Examples of interactions with the DS violating the Grice's maxims.

A. **User:** *The automaton accepts a language allowing words made of an odd number of 0s and 1s?* **DS:** *The language accepted by this automaton is made of zero or more words formed by a sequence of a triple of 1s followed by a pair of 0s.*
B. **User:** is q4 linked to q0? **DS:** *From q0 there is only one transition to q1 marked with 1.*
C. **User:** *There is a transition between q2 and q0?* **DS:** *No, there is no such transition. From q2 there is only one transition to q1 marked with 1.*

TABLE 6. Results obtained by VIP participants only, using ST and DS with the closed-ended questions – Average (standard deviation).

	Knowledge	Reasoning	All
$\mathbf{ST_{VIP}}$	4.43 (1.51)	**3.71** (1.11)	8.10 (0.41)
$\mathbf{DS_{VIP}}$	**4.86** (0.38)	3.57 (0.96)	**8.40** (0.37)

TABLE 7. Results obtained by VIP participants only using ST and DS with the open-ended questions about automata description – Average (standard deviation).

	Automata description
$\mathbf{ST_{VIP}}$	**1.86** (0.90)
$\mathbf{DS_{VIP}}$	**1.86** (0.38)

TABLE 8. Overall rating provided to ST and DS by VIP participants, on a scale of 1 ("very poor") to 5 ("very good") – Average (standard deviation).

	Overall rating
$\mathbf{ST_{VIP}}$	2.71 (0.76)
$\mathbf{DS_{VIP}}$	**4.57** (0.53)

Four different instruction-tuned models were specifically evaluated: ChatGPT-3.5,[13] LLama 2 7B [38], Tk-Instruct-11B [39] and Mistral 7B [40]. These models were used with their default settings. Additionally, both Llama 2 and Mistral were fine-tuned by using the corpus obtained during the experimentation described in Section V, in order to observe their performance in comparison to their non-fine-tuned versions. Starting from their base model, Low-Rank Adaptation (LoRA) [42] was performed. Llama 2 and Mistral were trained for two epochs each (all training details and hyperparameters are available in the appendix).

Below, the construction of the prompt, the experimental protocol, and the results are described. To construct a suitable prompt, Reference [41] suggests that the most effective prompts for instruction-tuned systems are those focused on reasoning, where the prompt instructs the model to explain the reasoning with a complex reasoning chain. This approach has been shown to improve their performance [43]. In this case study, given a question about an FSA, the reasoning prompt is built by providing the FSA in tabular format (ST), the question about the FSA, an explanation of how to read the ST, and the logical steps needed to answer the question. For example, the prompt regarding the question "What is the initial state?", consisted in the FSA as a table, the question, an explanation of how to read the table (e.g., identifying the cells in the ST with the symbol "→" indicating the initial state followed by the name of the state). The expected output would include the answer to the question along with the motivation leading to that answer.

To fine-tune the models, a prompt template was constructed based on the corpus, utilising question-answer pairs. The FSA was presented in a tabular format with explanations on how to interpret them. Additionally, the models were trained to answer user questions using responses from the rule-based DS.

For the experimental protocol, four typical questions about FSA that a student might ask a teacher were first selected. For each question, a corresponding reasoning prompt was created. The prompt was then tested in three settings: zero-shot, one-shot, and few-shot (with ten positive examples).[14] The four questions are as follows:

1) Based on the ST provided, is there a transition from q0 to q1?
2) Based on the ST provided, can you tell me how many transitions are there?
3) Based on the ST provided, can you tell me what is the accepted language?
4) Based on the ST provided, can you tell me which are the initial and final states?

As a result, 12 prompts were obtained by using different settings (zero-shot, one-shot, and few-shot) for each of the four questions. The examples were custom-built, and each prompt was executed once.

Three in-house volunteers, experts in FSA, were recruited. Two of the annotators hold a PhD in Computer Science, while

[13] https://openai.com/chatgpt - ChatGPT Sep. 25 2023.

[14] Some examples of prompts are reported in Tables 16–21. All the prompts are available at https://github.com/Reasoning-NLG-Unito/NoVAGraphs_

the third has a Master's degree in Computer Science. The three experts evaluated all the LLMs responses, separately assessing the correctness of LLMs' answers and motivations in a binary fashion: by marking 1 if the LLM response is correct, and 0 otherwise. A motivation was considered correct if it was sound and accurate. It is worth pointing out that the annotators were instructed to ignore the motivation, labelling it as inaccurate, if the answer itself was inaccurate. This restrictive criterion was motivated by the educational purpose of use.

The results obtained are shown in Tables 9 and 10. Table 9 displays the aggregated results assigned to the models responses in the three tested settings: zero-shot (*0s*), one-shot (*1s*), and few-shot (*fs*). The maximum score is 1, the minimum is 0. As highlighted in bold in Table 9, ChatGPT-3.5 achieved the best result in the one-shot setting, as it achieved the highest scores for answers (0.75) and motivations (0.67). However, when looking only at the answers, Llama 2 outperformed the other LLMs in the three settings. This performance is not consistent with the performance on the motivations (0.25 across the settings). Tk-Instruct-11B and Mistral received similar evaluations on the answers compared to ChatGPT-3.5, but, according to the annotators, they consistently provided inaccurate motivations. Notice that the fs performance is worse than 1s performance. This is posited to be due to the limitations of the LLM's context window size, which cannot accommodate the number of tokens in the examples. Regarding the fine-tuned models, both performed worse than their non-fine-tuned counterparts. Although this result is surprising, it is likely attributable to the characteristics of the corpus used for fine-tuning in both Llama 2 and Mistral. Specifically, the resulting corpus is quite small and carries the limitations inherent to interactions with rule-based systems, as the DS frequently fails to answer questions correctly, and there is a high repetition of questions and answers.

Table 10 reports the experts' agreement results using Krippendorff's Alpha [44] and Fleiss' Kappa [45]. The first row shows the overall agreement considering both the answers and the motivations. The second row focuses on the answers, while the last on the motivations. The three experts are in perfect agreement in asserting whether the model answers are correct or incorrect, and in almost perfect agreement considering both answers and motivations. A substantial agreement is achieved considering the soundness of motivations.

The results in Table 9 suggest that the tested LLMs in the different settings (i.e., both resorting to in-context learning and fine-tuning approaches) are not adequate to provide answers and motivations on FSA. The experts have commented that the models often provided incomplete motivations, sometimes made errors in the syntax and formalism of the FSA, and generated hallucinated information. This is especially problematic in an educational setting, where non-experts, such as students, may not be able to identify errors in responses that sound fluent and well-constructed. Additionally, it can be concluded that the fine-tuned models also performed worse, leading to the argument that there is a need to improve both the size and quality of the training corpus.

TABLE 9. Average results per LLM on zero-shot (*0s*), one-shot (*1s*) and few-shot (*fs*) settings and with fine-tuning.

Model	Shot	Answer	Motivation
GPT3.5	0s	0.25	0.00
	1s	**0.75**	**0.67**
	fs	0.25	0.17
Tk-Instruct-11B	0s	0.50	0.00
	1s	**0.75**	0.00
	fs	0.25	0.00
Llama 2	0s	**0.75**	0.25
	1s	**0.75**	0.25
	fs	0.50	0.25
Mistral	0s	**0.75**	0.00
	1s	0.25	0.00
	fs	0.25	0.00
Fine-tuned Llama 2	0s	0.25	0.00
	1s	**0.75**	0.17
	fs	0.50	0.17
Fine-tuned Mistral	0s	0.00	0.00
	1s	0.50	0.25
	fs	0.50	0.25

TABLE 10. Agreement among annotators.

Evaluation	Kripp.'s Alpha	Fleiss' Kappa
Answer+Motivation	0.91	0.92
Answer	1.00	1.00
Motivation	0.72	0.74

Table 11 reports a common error taking as example the Llama 2 response to the third prompt in the few-shot setting. The experts pointed out that the model—besides making a mistake in the formalism of the automaton by writing "(110)*)" in the motivation, and despite providing the correct accepted language (i.e., (110)*) in the answer—does not correctly recognise the transitions, thus creating inconsistency between the described transitions and the accepted language. In fact, Llama 2 described as the accepted language $(01 + 0)^*$, whereas the actual path described in the ST leading to (110)* is $q0 \xrightarrow{1} q2 \xrightarrow{1} q1 \xrightarrow{0} q0$.

VII. MAIN FINDINGS AND LIMITATIONS

Based on the results presented in the previous sections, the primary outcomes of this research are summarised as follows.

The first research question sought to determine whether a DS is generally more effective than an ST for acquiring knowledge about FSA. The investigation indicates that while users perform well with both methods, interaction with the

TABLE 11. Example of a few-shot prompt and a LLama 2 answer. The positive examples are not provided to make the prompt more readable.

Prompt

Definition: In this task you have to tell me what is the accepted language by the automaton. [...]
How to read the ST: [...]
positive examples [...]
Automaton [...] ST=<table><thead><tr><th> </th><th>0</th><th>1</th></tr> </thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr> </tbody></table>
Question: Based on the provided ST, can you tell me what is the accepted language? Please answer using the template "The accepted language is x." (where x is the accepted language you identified) followed by a motivation.

Output

The accepted language is $(110)*$.
The automaton can be drawn as a triangle, with three states - $q0$, $q1$, and $q2$. The initial state is $q0$, and the final state is $q2$. The transitions are as follows: From $q0$, with input 0, it moves to $q1$. From $q1$, with input 1, it moves to $q2$. From $q2$, with input 0, it moves back to $q0$. From $q2$, with input 1, it loops back to $q2$. The accepted language is $(110)*$) because from the initial state $q0$, it can accept the string 110 by following the path $q0 \rightarrow q1 \rightarrow q2 \rightarrow q0$. The string 110 can be accepted again by looping back to $q2$ from $q0$ with input 1, and then moving back to $q0$ with input 0. Therefore, the accepted language includes all strings of the form $(110)^*$, which means that any number of $1s$ can be appended to the beginning of the string 110.

TABLE 12. List of acronyms and their meaning.

Abbreviation	Description
AIML	Artificial Intelligence Markup Language
DS	Dialogue System
E-R	Entity Relationship
FSA	Finite State Automata
LLM	Large Language Models
NLP/G	Natural Language Processing/Generation
NLU	Natural Language Understanding
ST	State Table
UML	Unified Modeling Language
VIP	Visually Impaired People

TABLE 13. Accepted Language = (110)* State Table representation for the automaton with 3 states and 3 transitions.

State	0	1
$* \rightarrow$ q0	/	q2
q1	q0	/
q2	/	q1

TABLE 14. Accepted Language = (11100)* State Table representation for the automaton with 5 states and 5 transitions.

State	0	1
$* \rightarrow$ q0	/	q1
q1	/	q2
q2	/	q3
q3	q4	/
q4	q0	/

DS emerges as a viable alternative to the traditional ST-based approach for accessing and understanding FSA. Statistical analysis using a two-tailed non-parametric Wilcoxon signed-rank test shows that the performance difference between the DS and ST methods is statistically significant. This finding holds true for both VIP and non-VIP participants, highlighting the DS's effectiveness across different user groups. To our knowledge, our research is the first one that investigates this specific point.

To address the second research question, which aimed to identify the preferred method for VIP, the results and feedback from VIP participants were examined. Slightly divergent results were observed, with VIP participants showing better performance with either the DS or ST depending on the specific questions posed. It is posited that these variations may be due to VIP's specialised skills in navigating tabular representations, as also noted in [7] regarding comparisons of system interfaces. However, feedback from VIP participants revealed a clear preference for the DS, particularly regarding its ease of use. This suggests that the DS has the potential to offer a more effective and satisfying user experience for these users. Once again, to our knowledge, this analysis of VIP user experience has not been previously performed in the literature.

The third research question aimed to assess whether recent LLMs can address accessibility issues for VIP in an educational context. Despite the high performance of LLMs in many language understanding and generation tasks, as well as dialogue management, the tested models proved ineffective in accurately describing FSA. This inefficacy underscores significant limitations in accuracy and consistency of responses, presenting challenges in educational settings. This finding emphasizes the need for more explainable and manageable DS development approaches that offer precise control over content and interaction dynamics in these contexts. Based on current research, this is the first study to test two distinct LLMs in the specific context of teaching FSA adopting different fine-tuning and prompting strategies.

This study also identified several important limitations, primarily related to the interaction design of the DS and its evaluation. As discussed in previous sections, implementing a rule-based system poses issues related to scalability and adaptability across different contexts or knowledge domains. This is particularly evident with a pattern-matching paradigm, like the one used in this study, which does not ensure comprehensive coverage of lexical and interaction

TABLE 15. **Invitation letter distributed via mailing lists.**

Dear Esteemed Colleagues, We are a research group of Computer Science Department and Polin Laboratory at University of Turin. We need your contribution to test a dialogue system for exploring finite automata. You will interact with a dialogue system we developed with the accessibility goal in mind. Introducing the NoVAGraphS Research Experiment Our research team comprising Alessandro Mazzei, Luca Anselma, Pier Felice Balestrucci, Cristian Bernareggi, Elisa Di Nuovo and Manuela Sanguinetti, is thrilled to extend a heartfelt invitation to individuals with a basic knowledge of English and finite state automata. We need your valuable insights to evaluate our prototype system designed to enhance the accessibility of automata, a graphical structure usually taught in Computer Science courses which is not yet fully accessible for individuals with visual impairments. **What would you expect?** * Interact with the dialogue system: you will interact with the prototype dialogue system using a web interface to explore a simple automaton. * Explore another simple automaton using the state transition table: you will have access to a state transition table to access another simple automaton. * Fill a questionnaire in Google Form: we will ask you some questions about the two automata you explored using the above-mentioned methods and collect your feedback on the usability of the dialogue system. Rest assured, any personal information collected will remain completely anonymous. **Interested in participating or needing more information to decide?** To participate or to request more information, please contact pierfelice.balestrucci@unito.it

TABLE 16. **Example of Prompt 1 - zero-shot.**

prompt	**Definition**: In this task we ask you tell me if there is a transition from q0 to q1 based on the state table provided. Answer yes or no followed by a motivation. **How to read the ST**: How to read the transition table: $\rightarrow$ indicates if the state in that cell is initial; indicates if the state in that cell is final. Note that if you find these symbols in a cell, they does not correspond to a state but they give information about the state in that cell. The table contains a header (between $< thead > \ldots < /thead >$). The header in state transition tables contains a first empty cell, followed by as many cells as there are characters in the alphabet, one cell per character. In the body, the first column represents a specific state, and the following columns, if there are any transitions from that specific state, contain transitions from that specific state to other states marked by the character of the alphabet in the corresponding column. Complete the following example - Alphabet = 0,1. Accepted Language = (110)*. The automaton can be drawn as a triangle. Input=<table><thead><tr><th> </th><th>0</th><th>1</th></tr></thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr></tbody></table> **Question**: Is a transition from q0 to q1 based on the state table in the Input? Answer yes or no followed by a motivation.
output	...

TABLE 17. **Example of Prompt 1 - one-shot.**

prompt	**Definition**: In this task we ask you tell me if there is a transition from q0 to q1 based on the state table provided. Answer yes or no followed by a motivation. **How to read the ST**: How to read the transition table: $\rightarrow$ indicates if the state in that cell is initial; indicates if the state in that cell is final. Note that if you find these symbols in a cell, they does not correspond to a state but they give information about the state in that cell. The table contains a header (between $< thead > \ldots < /thead >$). The header in state transition tables contains a first empty cell, followed by as many cells as there are characters in the alphabet, one cell per character. In the body, the first column represents a specific state, and the following columns, if there are any transitions from that specific state, contain transitions from that specific state to other states marked by the character of the alphabet in the corresponding column. Positive Example 1 - <table><thead><tr><th> </th><th>0</th><th>1</th></tr> </thead><tbody><tr><th> * -> q0 </th><th> </th><th> q0 </th></tr></tbody></table> The ST in Example 1 indicates that q0 is both initial and final state. From q0 there is no transition with input 0 (empty cell in the middle column), and a transition to q0 with input 1 (because cell contains q0, in the last column). The answer to the question "is there a transition from q0 to q1?" is no, as no cell in the row containing all the transition from q0 contains only the state q1. Complete the following example - Alphabet = 0,1. Accepted Language = (110)*. The automaton can be drawn as a triangle. Input=<table><thead><tr><th> </th><th>0</th><th>1</th></tr></thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr></tbody></table> **Question**: Is a transition from q0 to q1 based on the state table in the Input? Answer yes or no followed by a motivation.
output	...

patterns, as noted in participants' feedback. Although this approach was chosen to test the working hypothesis, further refinements and adaptations will be necessary for future developments.

TABLE 18. Example of Prompt 1 - few-shot.

prompt	**Definition**: In this task we ask you tell me if there is a transition from q0 to q1 based on the state table provided. Answer yes or no followed by a motivation. **How to read the ST**: How to read the transition table: $\rightarrow$ indicates if the state in that cell is initial; indicates if the state in that cell is final. Note that if you find these symbols in a cell, they does not correspond to a state but they give information about the state in that cell. The table contains a header (between $<$ *thead* $>$ ... $<$ */thead* $>$). The header in state transition tables contains a first empty cell, followed by as many cells as there are characters in the alphabet, one cell per character. In the body, the first column represents a specific state, and the following columns, if there are any transitions from that specific state, contain transitions from that specific state to other states marked by the character of the alphabet in the corresponding column. Positive Example 1 - <table><thead><tr><th> </th><th>0</th><th>1</th></tr> </thead><tbody><tr><th> * -> q0 </th><th> </th><th> q0 </th></tr></tbody> </table> The ST in example 1 indicates that q0 is both initial and final state. From q0 there is no transition with input 0 (empty cell in the middle column), and a transition to q0 with input 1 (because cell contains q0, in the last column). The answer to the question "is there a transition from q0 to q1?" is no, as no cell in the row containing all the transition from q0 contains the string "q1". Positive Example 2 - <table><thead><tr><th> </th><th>0</th><th>1</th></tr> </thead><tbody><tr><th> * -> q0 </th><th> q0 </th> <th> </th></tr></tbody></table> The ST in example 2 indicates that q0 is both initial and accepting state. From q0 there is a transition to q0 with 0 as input (cell contains q0, in the middle column), and no transition with input 1 (empty cell in the last column). The answer to the question "is there a transition from q0 to q1?" is no, as no cell in the row containing all the transition from q0 contains the string "q1". ... Positive Example 10 - <table><thead><tr><th> </th><th>0</th><th>1</th><th>2 </th></tr></thead><tbody><tr><th> -> * q0 </th><th> q3 </th><th> </th><th> </th></tr><tr><th> q1 </th><th> q0 </th><th> </th><th> q3 </th></tr><tr><th> q2 </th><th> </th><th> q3 </th><th> </th></tr><tr><th> q3 </th><th> </th><th> q2 </th><th> q1 </th></tr></tbody></table> The ST in example 10 indicates that q0 is both the initial and final state. From the initial state q0 there is one trasition to q3 with 0 as input, and no transition with 1 or 2 as input. Then, from q1 there is a transition to q0 marked by 0, no transition marked by 1, and a transition to q3 marked by 2. From q2 there is no transition with input 0 and 2, and one transition to q3 with input 1. Finally, from q3 there is no transition with input 0 and a transition to q2 with input 1 and a transition to q1 marked by 2. The answer to the question "is there a transition from q0 to q1?" is no, as no cell in the row containing all the transition from q0 contains the string "q1". Complete the following example - Alphabet = 0,1. Accepted Language = (110)*. The automaton can be drawn as a triangle. Input=<table><thead><tr><th> </th><th>0</th><th>1</th></tr></thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr></tbody></table> **Question**: Is a transition from q0 to q1 based on the state table in the Input? Answer yes or no followed by a motivation.
output	...

TABLE 19. Example of Prompt 2 - zero-shot.

prompt	**Definition**: In this task we have to tell me how many transitions are in the automaton I provide you. Please answer using the template "The number of transition is n." (where n is the number of transitions you identified) followed by a motivation. **How to read the ST**: How to read the transition table: $\rightarrow$ indicates if the state in that cell is initial; indicates if the state in that cell is final. Note that if you find these symbols in a cell, they does not correspond to a state but they give information about the state in that cell. The table contains a header (between $<$ *thead* $>$ ... $<$ */thead* $>$). The header in state transition tables contains a first empty cell, followed by as many cells as there are characters in the alphabet, one cell per character. In the body, the first column represents a specific state, and the following columns, if there are any transitions from that specific state, contain transitions from that specific state to other states marked by the character of the alphabet in the corresponding column. Complete the following example - Alphabet = 0,1. Accepted Language = (110)*. The automaton can be drawn as a triangle. Input=<table><thead><tr><th> </th><th>0</th><th>1</th></tr></thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr></tbody></table> **Question**: Based on the ST provided in the Input, can you tell me what is the accepted language? Please answer using the template "The accepted language is x." (where x is the accepted language you identified) followed by a motivation.
output	...

Additionally, there is a need to expand the pool of VIP participants to achieve a more robust statistical analysis and obtain significant evidence within this user category. Recruitment difficulties were previously noted, especially due to the strict requirements for participation; nevertheless, gathering more feedback from VIP users is crucial to ensure the DS meets their information needs and expectations.

Finally, the limitations of using a closed model such as ChatGPT for the experiments described in Section VI are acknowledged. This model, accessible only through a restricted API, presents challenges related to result reproducibility. To address this issue, all prompts used in testing the models have been shared, to enhance transparency and provide a clearer understanding of the experimental methodology.

VIII. CONCLUSION

This article introduced a novel approach for exploring FSA in an accessible manner for VIP. Unlike HTML-based STs for accessing FSA, the DS approach provides a more natural interaction, as confirmed by positive user feedback. The study aimed to evaluate the effectiveness of this approach, and the preliminary results were promising.

A rule-based DS was implemented and tested against the HTML ST in an A/B test. The results demonstrated that exploring FSA via the DS significantly improved

TABLE 20. Example of Prompt 3 - zero-shot.

prompt	**Definition**: In this task we have to tell me what is the accepted language by the automaton. Please answer using the template "The accepted language is ..." followed by a motivation. **How to read the ST**: How to read the transition table: ⟶ indicates if the state in that cell is initial; * indicates if the state in that cell is final. Note that if you find these symbols in a cell, they does not correspond to a state but they give information about the state in that cell. The table contains a header (between <thead>...</thead>). The header in state transition tables contains a first empty cell, followed by as many cells as there are characters in the alphabet, one cell per character. In the body, the first column represents a specific state, and the following columns, if there are any transitions from that specific state, contain transitions from that specific state to other states marked by the character of the alphabet in the corresponding column. Complete the following example - Alphabet = 0,1. The automaton can be drawn as a triangle. Input=<table><thead><tr><th> </th><th>0</th><th>1</th></tr></thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr></tbody></table> **Question**: Based on the ST provided in the Input, can you tell me what is the accepted language? Please answer using the template "The accepted language is x." (where x is the accepted language you identified) followed by a motivation.
output	...

TABLE 21. Example of Prompt 4 - zero-shot.

prompt	**Definition**: In this task you have to tell me which are the initial and final states in the provided automata. Please answer by saying the names of the initial and final states followed by a motivation. **How to read the ST**: How to read the transition table: ⟶ indicates if the state in that cell is initial; * indicates if the state in that cell is final. Note that if you find these symbols in a cell, they does not correspond to a state but they give information about the state in that cell. The table contains a header (between <thead>...</thead>). The header in state transition tables contains a first empty cell, followed by as many cells as there are characters in the alphabet, one cell per character. In the body, the first column represents a specific state, and the following columns, if there are any transitions from that specific state, contain transitions from that specific state to other states marked by the character of the alphabet in the corresponding column. Complete the following example - Alphabet = 0,1. Accepted Language = (110)*. The automaton can be drawn as a triangle. Input=<table><thead><tr><th> </th><th>0</th><th>1</th></tr></thead><tbody><tr><th> * -> q0 </th><th> </th><th> q2 </th></tr><tr><th> q1 </th><th> q0 </th><th> </th></tr><tr><th> q2 </th><th> </th><th> q1 </th></tr></tbody></table> **Question**: Based on the ST provided in the Input, can you tell me which are the initial and final states? Please answer by saying the names of the initial and final states followed by a motivation.
output	...

participants' overall understanding compared to the ST. The study also analysed the advantages and disadvantages of the DS method through interaction analysis and collected a corpus of question-answer pairs, which will be published in the near future. Additionally, the inadequacy of using LLMs was highlighted, as tests with out-of-the-box ChatGPT, Llama 2, Tk-Instruct, and Mistral, as well as fine-tuning Llama 2 and Mistral, revealed limitations. Moreover, although LLMs have opened new possibilities for developing personalised assistive DS, they must also be carefully implemented to avoid unintended negative outcomes as well as ethical risks, as highlighted in [46].

The experiment results, which answer to our three research questions concerning the use of DSs for teaching FSA to VIP, shed new light on the applicability of dialogue systems as accessible technologies.

Future improvements to the rule-based DS are planned, including: (1) augmenting AIML patterns based on feedback received, (2) developing a scalable AIML rule engine capable of employing a more sophisticated pattern-matching scheme and implementing the information state dialogue model [47], (3) integrating the AIML engine with an LLM to enhance the variety of responses, and (4) expanding and improving the corpus obtained from the experiments to make it more effective for training LLMs.

Further development could also focus on expanding the target users to include students learning FSA. In this direction, the DS should be designed to proactively assess and address learning deficiencies.

NOTATION LIST

APPENDIX A

A. KNOWLEDGE QUESTIONS

1. How many states are there?

a) 1
b) 2
c) 3
d) 4
e) 5

2. What is the final state?

a) q0
b) q1
c) q5

3. How many transitions are there?

a) None
b) 3
c) 5
d) 8
e) 10

TABLE 22. Examples of some of the AIML Rules used by the DS.

Category	Pattern	Template
Greetings	HI	Hi. I'm the chatbot NoVAGraphS. I'm programmed to answer questions to describe graphical structures, and in particular, automata. Ask me to describe you in general Automaton 1 or a more specific question about it.
	HI *	`<srai>HI</srai>`
	HELLO	`<srai>HI</srai>`
	HELLO *	`<srai>HI</srai>`
Description	* LANGUAGE *	The language accepted by this automaton is made of zero or more words formed by a sequence of a triple of 1s followed by a pair of 0s.
	* LANGUAGE	<srai><star/> LANGUAGE <star index="2"/></srai>
	* ALPHABET	The alphabet accepted by this automaton is a binary alphabet made by 0s and 1s. If you want to know about the language accepted, ask me about it.
	* DETERMINISTIC *	This is a deterministic automaton.
	* REPRESENTED *	This automaton can be represented as a pentagon with its states arranged as vertices.
	* DRAWN	<srai><star/> REPRESENTED <star index="2"/></srai>
	* DRAWN *	<srai><star/> REPRESENTED <star index="2"/></srai>
	WHAT * ASK	You can ask me to describe you the automaton or a more specific question about it. For example, you can ask me about transitions or whether there is a path between two nodes.
	WHAT * ASK *	<srai>WHAT <star/> ASK</srai>
	* BRIEFLY *	This automaton accepts zero or more words in a binary alphabet formed by a sequence of a triple of 1s followed by a pair of 0s.
	* BRIEFLY	<srai><star/> BRIEFLY <star index="2"/></srai>
	HOW MANY STATES AND *	In this automaton there are 5 states and 5 transitions. From each state only one transition starts. Each state receives one transition.
	HOW MANY ARCS AND *	<srai>HOW MANY STATES AND <star/></srai>
	HOW MANY TRANSITIONS AND *	<srai>HOW MANY STATES AND <star/></srai>
	HOW MANY ARCS *	There are a total of 5 arcs. Try asking me if there is a particular pattern among them.
	* INITIAL * FINAL STATE	q0 is both the initial and final state.
	* PATTERN *	There is only one arc starting from each state. The first three in clockwise direction are marked with 1, the remaining two with 0.
	* PATTERN	<srai><star/> PATTERN <star index="2"/></srai>
Howto	* HELP *	Examples of questions may be: Briefly describe the automaton, How many states are there?, Is there an arc from q0 to q1?, How is the automaton represented? For more information click on Homepage tab.
	* HELP	<srai><star/> HELP <star index="2"/></srai>
Fallback	*	<random> - I don't know. Try asking something like, "Describe the automaton." - I don't know. Try asking something like, "Which is the initial state?" - I don't know. Try just asking something like, "How many states are there?" - I don't know. Try asking something like, "What is the accepted language?" - I don't know. Just try asking something like, "How is the automaton represented?" </random>

4. Is there a transition from q0 to q5?

a) Yes
b) No, but there is a transition from q0 to q2
c) No, but there is a transition from q2 to q0
d) No, but there is a transition from q0 to q4
e) No, but there is a transition from q4 to q0
f) None of the above

5. According to your understanding of Automaton 1, what is its optimal spatial representation? (Multiple-choice answers allowed)

a) Linear
b) Triangle
c) Square
d) Pentagon
e) Other …

B. REASONING QUESTIONS

1. The automaton accepts a language allowing words made of an odd number of 0s and 1s

a) True
b) False

2. It is possible to find a repetitive pattern in the transitions

a) True
b) False

3. If q2 is the final state, the language accepted by the automaton is the same

a) True
b) False

4. If q0 is the final state, the language accepted by the automaton is the same

a) True
b) False

5. If q1 were the final state, …

a) …the automaton would no longer be made of finite states.
b) …the automaton would not be deterministic.
c) …the automaton would allow words of length 1.
d) …the automaton would allow words of length 2.
e) …the automaton would not change.

C. FEEDBACK

1. Did you ask a question to which the dialogue system did not give you a satisfactory answer? If so, write below your question(s) and the expected answer(s)

2. How would you improve the interaction?

D. EXPERIMENTAL SETTINGS

The following hyperparameters were used for the LoRA adapter:

- R = 64
- α = 16
- no bias
- dropout: 0.05
- Target modules: Q-projections, K-projections, V-projections, O-projections, gate-projections

The adapter was loaded in 4-bit precision without employing double quantization.

Training was conducted on an A40 GPU with a per-device batch size of 4. Fine-tuning took approximately 20 minutes per model. Gradient checkpointing was utilized during the process.

The training comprised 2 epochs with a learning rate of 2e-5, using a linear scheduler. A warmup ratio of 0.3 was applied.

E. FSA

See Tables 13 and 14.

F. INVITATION LETTER

See Table 15.

G. LIST OF PROMPTS

See Tables 16–21.

H. AIML RULES

See Table 22.

REFERENCES

[1] D. Ahmetovic, C. Bernareggi, J. Guerreiro, S. Mascetti, and A. Capietto, "AudioFunctions.Web: Multimodal exploration of mathematical function graphs," in *Proc. 16th Int. Web Conf.*, May 2019, pp. 1–10, doi: 10.1145/3315002.3317560.

[2] J. Paladines and J. Ramirez, "A systematic literature review of intelligent tutoring systems with dialogue in natural language," *IEEE Access*, vol. 8, pp. 164246–164267, 2020, doi: 10.1109/ACCESS.2020.3021383.

[3] N. Nenkov, G. Dimitrov, Y. Dyachenko, and K. Koeva, "Artificial intelligence technologies for personnel learning management systems," in *Proc. IEEE 8th Int. Conf. Intell. Syst. (IS)*, Sep. 2016, pp. 189–195, doi: 10.1109/IS.2016.7737420.

[4] F. Clarizia, F. Colace, M. Lombardi, F. Pascale, and D. Santaniello, "Chatbot: An education support system for student," in *Proc. 10th Int. Symp. Cyberspace Saf. Secur.*, Amalfi, Italy. Cham, Switzerland: Springer, Oct. 2931, pp. 291–302, doi: 10.1007/978-3-030-01689-0_23.

[5] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, 3rd ed., London, U.K.: Pearson, 2007, pp. 171–183.

[6] A. Mazzei, M. Monticone, and C. Bernareggi, "Using NLG for speech synthesis of mathematical sentences," in *Proc. 12th Int. Conf. Natural Lang. Gener.*, 2019, pp. 463–472, doi: 10.18653/v1/w19-8658.

[7] P. Balestrucci, L. Anselma, C. Bernareggi, and A. Mazzei, "Building a spoken dialogue system for supporting blind people in accessing mathematical expressions," in *Proc. CEUR Workshop*, vol. 3596, 2023, pp. 1–8.

[8] N. Besse, S. Rosset, J. J. Zárate, E. Ferrari, L. Brayda, and H. Shea, "Understanding graphics on a scalable latching assistive haptic display using a shape memory polymer membrane," *IEEE Trans. Haptics*, vol. 11, no. 1, pp. 30–38, Jan. 2018, doi: 10.1109/TOH.2017.2767049.

[9] C. Bernareggi, C. Comaschi, G. Dalto, P. Mussio, and L. P. Provenza, "Multimodal exploration and manipulation of graph structures," in *Proc. 11th Int. Conf. Comput. Helping People Special Needs*, Linz, Austria. Berlin, Germany: Springer, 2008, pp. 934–937, doi: 10.1007/978-3-540-70540-6_140.

[10] C. Bernareggi, D. Ahmetovic, and S. Mascetti, "μGraph: Haptic exploration and editing of 3D chemical diagrams," in *Proc. 21st Int. ACM SIGACCESS Conf. Comput. Accessibility*, Oct. 2019, pp. 312–317, doi: 10.1145/3308561.3353811.

[11] A. F. Van scoy, D. McLaughlin, J. V. Odom, R. T. Walls, and M. E. Zuppuhaur, "Touching mathematics: A prototype tool for teaching pre-calculus to visually impaired students," *J. Modern Opt.*, vol. 53, no. 9, pp. 1287–1294, Jun. 2006, doi: 10.1080/09500340600618652.

[12] J. Moll and E.-L.-S. Pysander, ''A haptic tool for group work on geometrical concepts engaging blind and sighted pupils,'' *ACM Trans. Accessible Comput.*, vol. 4, no. 4, pp. 1–37, Jul. 2013, doi: 10.1145/2493171.2493172.
[13] W. Yu and S. Brewster, ''Evaluation of multimodal graphs for blind people,'' *Universal Access Inf. Soc.*, vol. 2, no. 2, pp. 105–124, Jun. 2003, doi: 10.1007/s10209-002-0042-6.
[14] R. Sarkar, S. Bakshi, and P. K. Sa, ''Review on image sonification: A non-visual scene representation,'' in *Proc. 1st Int. Conf. Recent Adv. Inf. Technol. (RAIT)*, Mar. 2012, pp. 86–90, doi: 10.1109/RAIT.2012.6194485.
[15] S. Mascetti, A. Gerino, C. Bernareggi, and L. Picinali, ''On the evaluation of novel sonification techniques for non-visual shape exploration,'' *ACM Trans. Accessible Comput.*, vol. 9, no. 4, pp. 1–28, Dec. 2017, doi: 10.1145/3046789.
[16] J. Su, A. Rosenzweig, A. Goel, E. de Lara, and K. N. Truong, ''Timbremap: Enabling the visually-impaired to use maps on touch-enabled devices,'' in *Proc. 12th Int. Conf. Hum. Comput. Interact. Mobile Devices Services*, Sep. 2010, pp. 17–26, doi: 10.1145/1851600.1851606.
[17] A. J. R. Godfrey, P. Murrell, and V. Sorge, ''An accessible interaction model for data visualisation in statistics,'' in *Proc. 16th Int. Conf. Comput. Helping People Special Needs*. Springer, 2018, pp. 590–597, doi: 10.1007/978-3-319-94277-3.
[18] S. Wu, J. Wieland, O. Farivar, and J. Schiller, ''Automatic alt-text: Computer-generated image descriptions for blind users on a social network service,'' in *Proc. ACM Conf. Comput. Supported Cooperat. Work Social Comput.*, Feb. 2017, pp. 1180–1192, doi: 10.1145/2998181.2998364.
[19] V. Sorge, M. Lee, and S. Wilkinson, ''End-to-end solution for accessible chemical diagrams,'' in *Proc. 12th Int. Web All Conf.*, May 2015, pp. 1–10, doi: 10.1145/2745555.2746667.
[20] B. G. Zapirain, A. M. Zorrilla, I. Ruiz, and A. Muro, ''Learning electronics using image processing techniques for describing circuits to blind students,'' in *Proc. 10th IEEE Int. Symp. Signal Process. Inf. Technol.*, Dec. 2010, pp. 156–160, doi: 10.1109/ISSPIT.2010.5711764.
[21] S. Chockthanyawat, E. Chuangsuwanich, A. Suchato, and P. Punyabukkana, ''Towards automatic diagram description for the blind,'' in *Proc. Int. Conv. Rehabil. Eng. Assistive Technol.*, 2017, pp. 1–4, doi: 10.13140/RG.2.2.11969.04961.
[22] H. K. Ault, J. W. Deloge, R. W. Lapp, M. J. Morgan, and J. R. Barnett, ''Evaluation of long descriptions of statistical graphics for blind and low vision web users,'' in *Proc. 8th Int. Conf. Comput. Helping People Special Needs (ICCHP)*. Berlin, Germany: Springer, 2002, pp. 517–526, doi: 10.1007/3-540-45491-8.
[23] I. Å. Krjanec, M. S. Edhi, and V. Demberg, ''Barch: An English dataset of bar chart summaries,'' in *Proc. 13th Lang. Resour. Eval. Conf.*, 2022, pp. 3552–3560.
[24] E. Di Nuovo, M. Sanguinetti, P. Balestrucci, L. Anselma, C. Bernareggi, and A. Mazzei, ''Educational dialogue systems for visually impaired students: Introducing a task-oriented user-agent corpus,'' in *Proc. Joint Int. Conf. Comput. Linguist., Lang. Resour. Eval.*, 2024, pp. 5507–5519.
[25] OpenAI et al., ''GPT-4 technical report,'' 2023, *arXiv:2303.08774*.
[26] H. Tian, W. Lu, T. On Li, X. Tang, S.-C. Cheung, J. Klein, and T. F. Bissyandé, ''Is ChatGPT the ultimate programming assistant—How far is it?'' 2023, *arXiv:2304.11938*.
[27] J. C. F. de Winter, ''Can ChatGPT pass high school exams on English language comprehension?'' *Int. J. Artif. Intell. Edu.*, pp. 1–16, Sep. 2023, doi: 10.1007/s40593-023-00372-z. [Online]. Available: https://link.springer.com/article/10.1007/s40593-023-00372-z#citeas
[28] T. Susnjak and T. McIntosh, ''ChatGPT: The end of online exam integrity?'' *Educ. Sci.*, vol. 14, no. 6, p. 656, Jun. 2024, doi: 10.3390/educsci14060656.
[29] C. K. Lo, ''What is the impact of ChatGPT on education? A rapid review of the literature,'' *Educ. Sci.*, vol. 13, no. 4, p. 410, Apr. 2023, doi: 10.3390/educsci13040410.
[30] J. Qadir, ''Engineering education in the era of ChatGPT: Promise and pitfalls of generative AI for education,'' in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, May 2023, pp. 1–9, doi: 10.1109/EDUCON54358.2023.10125121.
[31] E. Kasneci et al., ''ChatGPT for good? On opportunities and challenges of large language models for education,'' *Learn. Individual Differences*, vol. 103, Apr. 2023, Art. no. 102274, doi: 10.1016/j.lindif.2023.102274.
[32] M. F. McTear, Z. Callejas, and D. Griol, *The Conversational Interface*, vol. 6. Cham, Switzerland: Springer, 2016, doi: 10.1007/978-3-319-32967-3.
[33] E. Brulé, B. J. Tomlinson, O. Metatla, C. Jouffrais, and M. Serrano, ''Review of quantitative empirical evaluations of technology for people with visual impairments,'' in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, Apr. 2020, pp. 1–14, doi: 10.1145/3313831.3376749.
[34] V. Clarke and V. Braun, ''Teaching thematic analysis: Overcoming challenges and developing strategies for effective learning,'' *Psychologist*, vol. 26, no. 2, pp. 120–123, 2013.
[35] F. Wilcoxon, ''Individual comparisons by ranking methods,'' in *Breakthroughs in Statistics: Methodology and Distribution*. New York, NY, USA: Springer, 1992, pp. 196–202, doi: 10.1007/978-1-4612-4380-9_16.
[36] K. Proudfoot, ''Inductive/Deductive hybrid thematic analysis in mixed methods research,'' *J. Mixed Methods Res.*, vol. 17, no. 3, pp. 308–326, Jul. 2023, doi: 10.1177/15586898221126816.
[37] H. P. Grice, ''Logic and conversation,'' *Syntax Semantics*, vol. 3, pp. 45–47, Jan. 1975.
[38] H. Touvron et al., ''Llama 2: Open foundation and fine-tuned chat models,'' 2023, *arXiv:2307.09288*.
[39] Y. Wang et al., ''Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks,'' 2022, *arXiv:2204.07705*.
[40] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, ''Mistral 7B,'' 2023, *arXiv:2310.06825*.
[41] Y. Fu, H. Peng, A. Sabharwal, P. Clark, and T. Khot, ''Complexity-based prompting for multi-step reasoning,'' in *Proc. 11th Int. Conf. Learn. Represent.*, 2022.
[42] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, ''LoRA: Low-rank adaptation of large language models,'' 2021, *arXiv:2106.09685*.
[43] A. Addlesee, W. Sieińska, N. Gunson, D. Hernández Garcia, C. Dondrup, and O. Lemon, ''Multi-party goal tracking with LLMs: Comparing pre-training, fine-tuning, and prompt engineering,'' 2023, *arXiv:2308.15231*.
[44] K. Krippendorff. (2011). *Computing Krippendorff's Alpha-Reliability*. [Online]. Available: https://www.asc.upenn.edu/sites/default/files/2021-03/Computing
[45] J. L. Fleiss, ''Measuring nominal scale agreement among many raters,'' *Psychol. Bull.*, vol. 76, no. 5, pp. 378–382, Nov. 1971, doi: 10.1037/h0031619.
[46] O. Pierrès, M. Christen, F. M. Schmitt-Koopmann, and A. Darvishy, ''Could the use of AI in higher education hinder students with disabilities? A scoping review,'' *IEEE Access*, vol. 12, pp. 27810–27828, 2024, doi: 10.1109/ACCESS.2024.3365368.
[47] D. R. Traum and S. Larsson, ''The information state approach to dialogue management,'' in *Current and New Directions in Discourse and Dialogue*. Dordrecht, The Netherlands: Springer, 2003, pp. 325–353, doi: 10.1007/978-94-010-0019-2.

PIER FELICE BALESTRUCCI received the B.S. degree in computer science from the University of Bari, Italy, in 2019, and the M.S. degree in computer science from the University of Turin, Italy, in 2022, where he is currently pursuing the Ph.D. degree in natural language processing.

ELISA DI NUOVO received the M.A. degree in foreign languages and linguistics from the University of Turin, where she is currently pursuing the Ph.D. degree in digital humanities. She defended her thesis in 2022, marking the beginning of her research journey into computational linguistics. She is currently a Research Project Officer with the Joint Research Centre, European Commission. Her research interests include text mining and sentiment analysis, corpus linguistics, learner corpus research, syntactic parsing, and dialogue.

MANUELA SANGUINETTI received the Ph.D. degree in computer science from the University of Turin, in 2016. She is currently a non-tenured Assistant Professor with the Department of Mathematics and Computer Science, University of Cagliari, where she has been involved in a project funded by the National Reform and Resilience Plan (PNRR). Her research interests include the development of linguistic resources to enhance language understanding and processing. She has been involved in a wide range of research collaborations regarding the study of task-oriented conversational agents, hate speech and stereotype detection, and multilingualism.

LUCA ANSELMA received the M.S. and Ph.D. degrees in computer science from the University of Turin, Italy, in 2002 and 2006, respectively. From 2006 to 2022, he was an Assistant Professor with the University of Turin, where he has been an Associate Professor with the Department of Computer Science, since 2022. He is the author of more than 70 papers published in international journals and conferences. His research interests include temporal relational databases, temporal reasoning, health informatics, natural language processing, and generation.

CRISTIAN BERNAREGGI received the Ph.D. degree in computer science, in 2006. From 2004 to 2009, he conducted research in human–computer interaction in various European projects on accessibility to science with digital tools. Since 2007, he has been a technical scientific collaborator in computer science for the University of Milan. From 2011 to 2018, he participated in the foundation and growth of the spin-off EveryWare Technologies, for the development of assistive technologies on mobile devices. Since 2018, he has been collaborating with the Polin Laboratory, University of Turin, on STEM accessibility for people with disabilities.

ALESSANDRO MAZZEI received the degree in physics from the University of Naples Federico II, Naples, Italy, in 2000, and the Ph.D. degree in computer science from the University of Turin, Turin, Italy, in 2005. He has been an Associate Professor of computer science with the University of Turin, since 2022. He has authored more than 100 papers published in international journals and international refereed conferences. His research interests include the areas of parsing, natural language generation, and dialogue systems.

Open Access funding provided by ‘Università degli Studi di Torino’ within the CRUI CARE Agreement