Computer Science and Information Systems 18(4):1271–1286 https://doi.org/10.2298/CSIS200731020C

Conversational Agent for Supporting Learners on a MOOC on Programming with Java

Cristina Catalán Aguirre, Nuria González-Castro, Carlos Delgado Kloos, Carlos Alario-Hoyos, and Pedro J. Muñoz-Merino

Universidad Carlos III de Madrid,
Av. Universidad, 30, 28911 Leganés, Madrid
{crcatalan@pa, nurigonz@db, cdk@it, calario@it, pedmume@it}.uc3m.es

Abstract. One important problem in MOOCs is the lack of personalized support from teachers. Conversational agents arise as one possible solution to assist MOOC learners and help them to study. For example, conversational agents can help review key concepts of the MOOC by asking questions to the learners and providing examples. JavaPAL, a voice-based conversational agent for supporting learners on a MOOC on programming with Java offered on edX. This paper evaluates JavaPAL from different perspectives. First, the usability of JavaPAL is analyzed, obtaining a score of 74.41 according to a System Usability Scale (SUS). Second, learners' performance is compared when answering questions directly through JavaPAL and through the equivalent web interface on edX, getting similar results in terms of performance. Finally, interviews with JavaPAL users reveal that this conversational agent can be helpful as a complementary tool for the MOOC due to its portability and flexibility compared to accessing the MOOC contents through the web interface.

Keywords: Conversational Agent, Computer Science, MOOC, Programming, Java.

1. Introduction

MOOCs (Massive Open Online Courses) are still one of the most important trends in online learning, allowing the interaction among learners with different backgrounds and from different origins. MOOCs have contributed to expand the access to quality content, especially in engineering and computer sciences as a good number of MOOCs are framed within these two areas of knowledge [1]. Moreover, some of the MOOCs are instructed by renowned teachers giving learners the opportunity to satisfy their appetite for learning even if they are not enrolled in any traditional university.

Nevertheless, MOOCs present important issues. For example, the fact that the courses are open, with no enrolment fees, goes hand in hand with a low commitment from a good number of participants [2]. Completion rates are usually below 10% of the enrolled learners [3]. Although this is not always a problem, since there are learners who take MOOCs just to explore the content they are interested in, there are some other learners who drop out because their level (or background) is not sufficient or because they are not able to take an online course autonomously. When it comes to engineering and computer sciences MOOCs, apart from the above-mentioned issues, learners have to struggle with the intrinsic difficulty of the contents.

Numerous studies have focused on analyzing dropout rates, especially in MOOCs on topics related to engineering and computer sciences, with the ultimate aim to enhance the course design or to introduce interventions that can serve to reduce these dropout rates [4][5]. For example, it is possible to detect if a learner is going to leave a MOOC earlier through the identification of behavioral patterns [6][7] and react by giving feedback or specific advice to that learner. For example, authors in [8] discuss the use of a mentoring program to motivate learners to not giving up before the end of the course, thus overcoming the limitation related to the lack of teacher support in MOOCs; however, this solution faces again the problem of scalability in MOOCs with a very large numbers of learners. In contrast, authors in [9] propose the use of conversational agents, eliminating the human component in order to better scale up, as a support to MOOC learners. The idea behind the use of conversational agents is to improve the learning impact through dialog (dialog learning) since the participation in a conversation gives learners a more active role in the learning activity they are doing [10]. However, the use of conversational agents in the learning process in MOOCs has not been deeply examined [11]. In consequence, the effectiveness and usability of conversational agents in MOOCs has yet to be evaluated.

A pioneering example of conversational agent to support learners enrolled in MOOCs is JavaPAL, for which a first prototype was briefly introduced in a "computers in education" conference [9]. JavaPAL is a voice-based conversational agent designed to support learners who are taking the MOOC on "Introduction to Programming with Java" deployed on the edX platform. The objective of JavaPAL is to facilitate the study and revision of concepts related to Java programming using dialog, providing definitions on key concepts and asking some related questions to the learners. Although JavaPAL operates independently from the MOOC (as a standalone conversational agent), all the concepts for which JavaPAL provides definitions, and all the questions asked by JavaPAL are taken (and adapted in the case of some questions) directly from the above-mentioned MOOC. JavaPAL is aimed at accompanying learners through the learning process and at serving as reinforcement of the basic concepts that learners need to grasp. Nonetheless, the usability of a conversational agent such as JavaPAL and its effect in the learning experience is still unknown [11]. Moreover, the differences between learners' interaction through a conversational agent and the traditional MOOC web interface have yet to be analyzed when it comes to reviewing key concepts and answering questions.

Thus, this paper aims to shed some light on the usability of question-driven conversational agents to support learners enrolled in MOOCs, and the comparison of learners' performance when answering questions through web and conversational agent interfaces, all this using the JavaPAL as an example case. Therefore, the research questions of this paper are:

RQ1: Can a question-driven conversational agent fulfil learners' expectations in terms of usability?

RQ2: Does the use of a conversational agent affect learners' performance when answering questions in MOOCs?

RQ3: What are the differences learners and teachers find between the use of a MOOC web interface and a conversational agent interface when reviewing concepts and answering questions?

The structure of the paper is as follows. Section 2 provides the background on conversational agents both text-based (chatbots) and voice-based, focusing on their application in education. Section 3 describes the methodology used to conduct this study, including a description of the conversational agent used, the process of collecting and analyzing the data and the tools used. The results of the analysis are presented in Section 4, answering the research questions. Section 5 draws the conclusions of this research work.

2. Related Work

The best way to express interest, wishes or queries directly and naturally to computers is by speaking (voice-based dialog) or typing (text-based dialog) since these kinds of communication facilitate Human Computer Interaction [12]. Conversational agents are dialog systems that not only conduct natural language processing but also respond automatically using human language [13]. Chatbots are defined as computer programs that mimic the human interaction by using a text-based conversational agent to provide the interaction [14]. Consequently, both text-based and voice-based conversational agents arise as the most appropriate technology to use to fulfil the extension of human communication with computers.

The increase in the presence of chatbots in society is such that there is an estimation of 80% of enterprises using them, and around 40% of companies having used one or more assistants or AI-based chatbots over mobile devices in 2019 [15]. However, these programs are not new at all: Eliza, the first chatbot, was created in 1964. Eliza was a textual chatbot that used simple keyword matching techniques to match user input: it was designed to simulate a psychotherapist [16]. Another example of chatbot is A.L.I.C.E., which was developed using Artificial Intelligence Markup Language (AIML). AIML consists of objects that contain topics and categories. Each category represents a rule to match a user input to generate an output. This match is based on the internal templates from A.L.I.C.E. [17].

It is important to notice that, in their early stages of development, chatbots were not intelligent systems since they provided some pre-programmed questions and gave specific and predetermined responses. Jia [18] highlights the idea of users being upset with the responses provided by basics chatbots, since their pattern-matching system can be considered insufficient to be used in a real conversation. However, with the development of artificial intelligence, conversational agents have the potential to learn and assume the role of humans in some areas, including education [11]. Thus, systems that can learn from their previous experiences using AI, like Edwing.ai [19], which can elaborate more personalized responses, can have a wider adoption.

The use of chatbots in education, for example to raise questions that can be answered by students, could help teachers detect weaknesses in their students, as well as identify concepts and topics which pose a greater challenge [20]. This idea can be particularly useful in engineering and computers sciences due to the usual complexity of the key concepts in these areas of knowledge. Furthermore, chatbots can give support to each individual student since they are in position to acknowledge strengths, abilities or interests, and encourage learners to be more independent and engaged. In addition, the development of new Machine Learning techniques has led to an important rise of a new generation of conversational agents for education, such as the one presented in [21], which relies on a Naïve Bayesian classifier to answer questions posted by students as if it were a virtual teacher. With the development of Machine Learning techniques and the improvement of Natural Language Understanding, conversational agents are expected to enhance their characteristics simplifying the communication between user (learner) and (conversational) agent.

Design strategies to build up conversational agents for education are diverse. Particularly, in e-learning conversational agents range from simple text-based forms to voice-based. All of them should share the same objective: acting as a partner for the students and taking a non-authoritative role in a social learning environment. Each design strategy reflects one of the possible conversational systems: 1) simple text-based forms in which users type their responses; 2) embodied conversational agents capable of displaying emotions and gestures; and 3) voice input and output systems able to synthesize text to speech and vice versa [22].

Intelligent tutoring systems (ITSs) have been widely studied for natural dialog in education, with some ITSs being framed under the definition of conversational agents. One example of ITS for education that can also be defined as a conversational agent is Adele, a pedagogical text-based agent to support learners in web-based educational simulations, such as medical simulations [23]. Another example ITS which acts as a text-based chatbot is TutorBot [24]. TutorBot follows a question/answer schema in which the learner can retrieve information from a knowledge source using natural language. Finally, one voice-based example of ITS is AutoTutor [25], which is able to keep conversations with humans in natural language and incorporates strategies of human tutors previously identified in human tutoring protocols [25].

Although the existence of conversational agents is far from being new, their application within MOOCs is still uncertain. Authors in [26] present a first approach on the use of conversational agents in MOOCs, although these authors only present the basis for the development of a conversational agent, but they do not actually implement it. Another example of conversational agent developed to support MOOC learners through dialog is Bazaar [27], a text-based chat tool that provides synchronous interaction to learners within MOOCs. Another example of the use of conversational agent in MOOCs is QuickHelper, whose aim is to help learners to reduce their reluctance to ask questions and increase the number of questions posted in the forums [11]. These pioneering conversational agents base their conversation with the MOOC learner on text rather than on voice so there is still a research gap on voice-based conversational agents to support MOOC learners.

Apart from the previous examples, conversational agents are becoming more important in online education and blended learning since the participation of conversational agents during a peer communication reinforces the knowledge about the topic by activating relevant cognitive activity [26]. Consequently, conversational agents in e-learning can also help to improve peer to peer interaction [26]. Furthermore, the integration of conversational agents in MOOCs may trigger productive interaction in group discussion, increase the engagement and the commitment of MOOC learners and, therefore, reduce the overall dropout rates [11]. In this line, authors in [9] introduced a first prototype of a voice-based conversational agent called JavaPAL, which enables the possibility of reinforcing the knowledge of the contents of a MOOC on “Introduction to

Programming with Java" using voice dialog. Since Java programming is one of the basics subjects of many engineering or computer science degrees, JavaPAL is meant to help MOOC learners in the process of learning the key concepts from the above-mentioned MOOC by reviewing these concepts and posing multiple choice questions to the learner.

Thanks to the development and the application of Natural Language Understanding, conversational agents are expected to improve their characteristics and functionalities in the next years. This can lead to an improvement in online education, especially within MOOCs, since conversational agents may allow learners to get support at any place by using their mobile phones, tablets or devices such as Google Home or Alexa. Particularly, MOOCs with more complex content, such as those related to engineering or computer sciences, can benefit from the advantages of conversational agents to improve the lack of the support they face. Consequently, some important problems MOOCs are currently facing can be addressed and blended education can be open to a new paradigm of opportunities in which conversational agents can play a major role.

3. Methodology

A mixed-methods design [28] was applied in this research work. More specifically, the mixed-methods sequential explanatory design was followed. First, quantitative data was collected through the SUS (System Usability Scale) questionnaire [29] and logs were obtained from the use of the conversational agent; these quantitative data served to answer the first two research questions. Then, qualitative data was collected through interviews with users of the conversational agent; these qualitative data served to answer the third research question. A controlled group of 39 users was selected to evaluate JavaPAL. The members of this group did not correspond to actual learners enrolled in the MOOC JavaPAL supports, but to a number of pre-selected users with several backgrounds (students, teachers, researchers) whose mission was to evaluate JavaPAL. All these users shared previous knowledge on the topic of the MOOC. More specifically, 39 users tested JavaPAL and subsequently filled in the SUS regarding the usability of the conversational agent (RQ1), and 15 of them participated later in a quasi-experimental design aimed at comparing JavaPAL and the MOOC web interface when answering questions, which resulted in the collection of evidences through logs (RQ2) and interviews (RQ3).

Next, there is an overview of JavaPAL, the conversational agent used in this research. The instruments for data collection and data sources used are further explained right after.

3.1. JavaPAL

JavaPAL is a voice-based conversational agent developed at Universidad Carlos III de Madrid with the aim of supporting learners enrolled in the MOOC on "Introduction to Programming with Java", which is deployed in the edX platform. JavaPAL operates as a standalone conversational agent and was implemented using the natural language understanding platform DialogFlow and runs on Google Assistant, so it can work on any device that supports Google Assistant (e.g., smartphone, Google Home, etc.) (more technical details can be found in [9]). In addition to supporting voice-based interaction as the main mode of operation, JavaPAL also supports text-based interaction if the user wants it. JavaPAL supports three operation modes (quiz, review, and status) and to enter any of these three operation modes the learner must start the conversation with JavaPAL and indicate the desired operation mode.

Quiz mode. This mode asks questions to the learner. These questions are taken from the MOOC on "Introduction to Programming with Java" and are arranged according to the modules of the MOOC (five modules). When the learner accesses the quiz mode, s/he indicates the module from which s/he wants to receive questions and JavaPAL provides random questions from that module. The total number of consecutive questions asked by JavaPAL is predefined to three, but this can be configured by the learner in the status mode. This means that the learner receives three questions of the same module before s/he can change to another module. In this operation mode, the conversational agent takes the initiative of the communication since it asks questions and the learner answers them. The questions are designed in a way that the learner only needs to give a word (options from "a" up to "e" for multiple choice questions or true/false) to make it easier to remember each option.

Review mode. This mode provides learners with the definition of the key concepts addressed in the MOOC. In this operation mode, the learner takes the initiative of the communication since s/he indicates JavaPAL the concept s/he wants to revise (e.g., "class", "object", "method", etc.) and then JavaPAL provides the corresponding definition. If the definition of the concept is not available, then JavaPAL offers the definition of another related concept. Moreover, besides offering the definition of the concept indicated by the learner, JavaPAL can suggest other related concepts to provide their definitions based on an ontology that relates concepts of Java programming [30].

Status mode. In this mode, the learner can check his/her performance during the quiz mode and change some settings, such as nickname or number of consecutive questions.

JavaPAL has been developed using an iterative approach. Prototypes have been developed and tested with real users, and aspects to be improved have been detected through surveys and interviews with the users, and subsequently implemented in the following prototypes.

3.2. Instruments and Data Sources

System Usability Scale (SUS). Standardized usability questionnaires were analyzed in a first phase. After this analysis, SUS (System Usability Scale) was chosen as it is more reliable and detects differences at smaller sample sizes than other questionnaires, and the number of items to be assessed is not very large (10 items, 5 options per item), which facilitates data collection [29]. The measurement of the validity has been calculated using Cronbach's alpha [31]. **39 answers** to the SUS questionnaire were obtained from users of JavaPAL. These answers came from users with different knowledge about Java, mixing students, teachers and researchers. SUS allowed to gather the information needed to answer *RQ1*.

Logs. JavaPAL has been designed to collect logs on users' interaction, users' performance (understood as the correct/incorrect answers in the quiz mode) and the concepts learners ask for (in the review mode). **15 users** participated in a quasi-experimental design aimed at collecting data through logs. These users had previous experience with Java programming and were divided into two groups. The first group was asked to use the conversational agent first and then the MOOC web interface, in both cases to answer questions related to the MOOC contents. The second group was asked to use the MOOC web interface first and then the conversational agent, again to answer questions related to the MOOC contents. The objective was trying to detect differences between both types of interfaces from the learner's performance. It is worth to mention that some questions had to be adapted for the conversational agent to be concise enough and easy to remember by the learners using voice interaction. Some example of these adaptations can be seen in Appendix A. JavaPAL logs allowed to gather the information needed to answer *RQ2*.

Interviews. An interview was conducted with the **15 users** after they completed the interaction with the two types of interfaces. The process followed to collect and analyze the data from the interviews was: 1) the interviews were recorded; 2) a transcription of the recordings was generated; 3) an analysis of the content of each interview was done using a codification technique. It is worth to mention that all conversations took place in the participants' native language (Spanish). Therefore, the content has been translated trying not to lose the significance of the words and expressions used. The questions asked during the interviews can be seen in Appendix A. The interviews allowed to gather the information needed to answer *RQ3*.

4. Results

This section answers the three research questions (RQs) of this work based on the data collected from: 1) SUS (RQ1); 2) Logs (RQ2); and 3) Interviews (RQ3).

4.1. Can a Question-driven Conversational Agent Fulfil Learners' Expectations in Terms of Usability? (RQ1)

The first step to follow while calculating the scores for SUS is to figure out the contribution of each question, whose range has to be between 1 (strongly disagree) and 5 (strongly agree). The *SUS Score* is calculated as $(X+Y)*2.5$, where

- X = Sum of the points for all odd-numbered questions – 5
- Y = 25 – Sum of the points for all even-numbered questions

The final score, *SUS Score*, is a number between 0 and 100. Once the *SUS Score* was computed for the 39 cases of the sample, the final scores ranged from 35 to 97.5 points out of 100 as it can be seen in the boxplot in Figure 1. Half the users scored JavaPAL within the range from 65 to 85. The median of the all scores is 77.5. The boxplot shows that the sample is positively skewed, as it is in the high part of the graph. The mean of the scores is 74.71 and the standard deviation is 16.479.

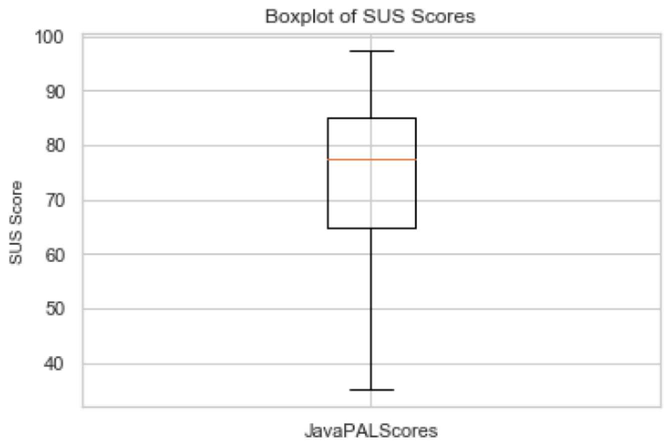

Fig. 1. SUS Global Scores boxplot for JavaPAL (median 77.5, mean 74.71).

SUS Score can also be converted into percentile ranks as indicated by Lewis and Sauro [32]. This percentile rank gives an idea of the usability of JavaPAL in relation to other products in a database. A score of 74.71 corresponds to a 70% of percentile rank, meaning that JavaPAL can be considered more usable than 70% of the products in the database [33] and less usable than 30 % of them. By definition, if a product has a percentile higher than 50% is considered to be above average.

Considering the scale by Bangor [29] (see Figure 2), it is possible to convert the SUS Score into grades, getting JavaPAL a C in this case. Moreover, in the Acceptability Ranges JavaPAL obtains "Acceptable", while in the Adjective Ratings JavaPAL obtains "Good". Bangor concludes that these results should be used together to create clearer pictures of products related to their overall usability [29].

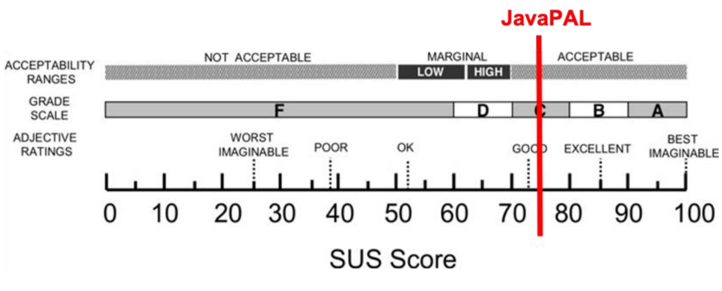

Fig. 2. SUS Bangor scale [29] and SUS Score for JavaPAL (mean value).

Another interpretation of SUS suggests dividing the items in two subscales: 1) "Learnability" (items 4 and 10); and 2) "Usability" (items 1-3 and 5-9) [33]. In order to compare the two subscales, the 2-item subscale should add the value of its items and multiply the results by 12.5, while the 8-item subscale should add the value of its items and multiply the result by 3.125. Figure 3 shows the results for "Learnability", while Figure 4 shows the results for "Usability". In the case of SUS Learnability Score the mean value is 83 and the median 87.5 (there is an outlier in 0); the results are positively skewed. In the case of SUS Usability Score the mean value is 72.51 and the median 75; Q1 and Q3 are 57.81 and 87.5, respectively, indicating that the results are also positively skewed.

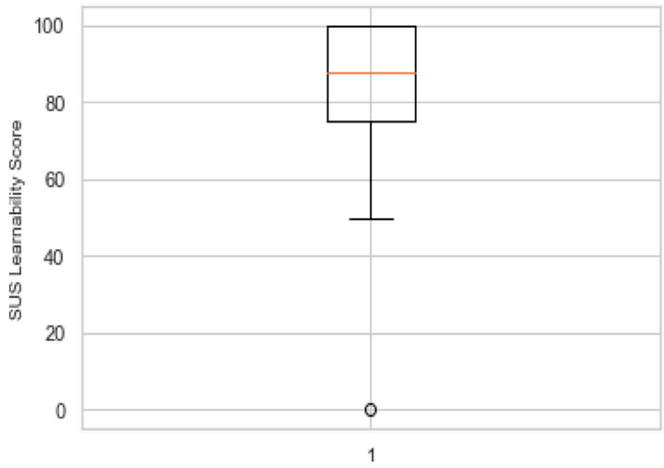

Fig. 3. SUS Learnability Score boxplot for JavaPAL (median 87.5, mean 83).

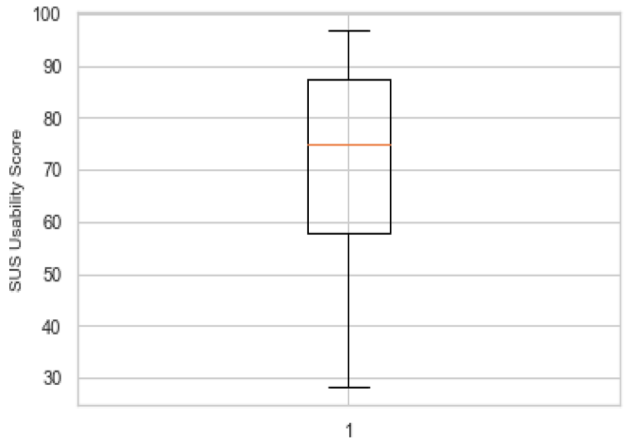

Fig. 4. SUS Usability Score boxplot for JavaPAL (median 75, mean 72.51).

All in all, regarding RQ1 it is possible to state that a conversational agent such as JavaPAL can provide a good usability to learners, including from the perspectives of both learnability and usability.

4.2. Does the use of a conversational agent affect learners’ performance when answering questions in MOOCs? (RQ2)

Logs were used to analyze the number of correct and incorrect answers provided by the participants (N=15) in the quasi-experiment. The number of correct and incorrect answers is the indicator used to measure learners' performance. Group 1 used the conversational agent first and then the MOOC web interface. Group 2 used the MOOC web interface first and then the conversational agent. Table 1 summarizes the results obtained

Table 1. Correct answers (mean and standard deviation) in Groups 1 and 2.

	Conversational agent	MOOC web interface
Group 1		
Mean correct answers	82.54%	74.6%
Standard deviation	15.83	8.32
Group 2		
Mean correct answers	89.76%	74.31%
Standard deviation	9.03	13.6

Group 1 members obtained a mean value of 74.6% of correct answers (SD=8.32) using the MOOC web interface, while Group 2 members obtained a mean value of 74.31% of correct answers (SD=13.6). In the case of the conversational agent, Group 1 obtained 82.54% of correct answers (SD=15.83), while Group 2 members obtained a mean value of 89.76% of correct answers (SD=9.03). In both cases the learners using the conversational agent obtained a higher percentage of correct answers on average.

A Mann-Whitney test of the difference between Group 1 and Group 2 was also applied. The confidence interval (-22.22, 5.5) was obtained with 95% confidence. This means that, in the worst-case scenario, and with a 95% of confidence, the use of conversational agent would decrease the learners’ performance by 5.5% (in terms of correct answers) when comparing with learner’s performance using the MOOC web interface.

All in all, regarding RQ2, it is possible to state that the use of the conversational agent like JavaPAL does not mean a worse learner’s performance, measured through the number of correct answers. Nevertheless, it is important to be cautious about these result as in some cases the questions presented to learners in the conversational agent had to be simplified (question and/or its answers) in order to make the question/answers easier to remember.

4.3. What are the differences learners and teachers find between the use of a MOOC web interface and a conversational agent interface when reviewing concepts and answering questions? (RQ3)

Interviews with 15 users of JavaPAL served to collect qualitative data with the aim to gain insights on the differences between the traditional MOOC web interface and the conversational agent interface when reviewing concepts and answering questions related to the MOOC.

One of the main advantages of JavaPAL highlighted by interviewees is the *immediate feedback*. This aspect was pointed out by *8 interviewees*. For example, Users 5 and 12 indicated that immediate feedback makes the conversational agent “more engaging”. User 6 added that thanks to the immediate feedback students could learn more because they would be more aware of their mistakes. Nevertheless, immediate feedback is a feature that can also be obtained through the MOOC web interface if the questions are configured properly. Nonetheless, it is easier (and faster) to ask the conversational agent to define a specific concept (in the review mode) than to search for the concept definition using the MOOC web interface.

Another positive aspect of JavaPAL highlighted by interviewees is the possibility of interacting with the conversational agent directly from the *mobile phone*. This aspect was pointed out by *6 interviewees*. For example, User 2 indicated that answering questions using the “mobile phone is faster than the mouse” in the web interface as it is the case when comparing voice-based interaction with text-based (or click-based) interaction. This same fact was also supported by three additional users. Users 1 and 5 also pointed out the benefits of using the conversational agent in the mobile phone “while travelling or commuting”. In addition, Users 3 and 7 also argued that the use of the conversational agent in the “mobile phone improves accessibility”. In contrast, User 1, for example, believed that the “web interface allows you to have a more general vision of the questions” you must answer unlike in the case of the conversational agent, and that the web interface allows for “more complex questions”. User 4 also indicated that, in general, people are “more used to using the web interface”, and that the web interface is “easier to use than the conversational agent”, which requires a certain learning curve. This idea was also reinforced by Users 1, 8 and 13. In contrast, User 3 claimed the opposite saying that more people interact through the mobile phone than through web interfaces.

The limitations of the conversational agent according to the interviewees are diverse. For example, User 4 believed that the conversational agent “cannot substitute the use of MOOCs” and it has to be seen as “a complementary tool”. User 13, for example, stated that it is “more complicated to type with the mobile phone than using the web interface”. However, four users believed that were no disadvantages in the case of the conversational agent. When it comes to the web interface, Users 7 and 10 identified as a drawback that there is no conversational interface. Users 8 and 9 both agreed on the fact that the web interface is “more monotonous” than the conversational agent and, in general, they all believe that the web interface is “more rigid” than the conversational agent.

Regarding preferences, User 8 indicated that he would always use the conversational agent. Users 1, 2 and 5 claimed that they would use the conversational agent while traveling instead of the web interface. The remaining users believed that the conversational agent would be preferable in case there was no access to the web interface, or when the content of the MOOC is highly theoretical. In contrast, Users 3 and 8 expressed a preference for the web interface. User 12 indicated a preference for the web interface only when having access to a computer, while User 15 indicated this preference for the web interface when accessing the main material to study.

Finally, in terms of learning, 6 interviewees believed that they would learn more with the conversational agent than with the web interface. For example, Users 8 and 11 believed that the conversational agent is “more engaging” so it would be easier to learn with it. However, 4 interviewees believed that they would obtain the same outcome when using the conversational agent and the web interface. On the contrary, 3 interviewees believed that the MOOC web interface would be better to learn because it has the videos and the questions interwoven and this is much better for those students with zero knowledge about Java programming.

Comments from interviewees were quite polarized regarding the preference between the use of a conversational agent like JavaPAL and the MOOC web interface. Regarding RQ3, and after assessing the advantages and disadvantages mentioned by the interviewees, it is possible to conclude that a conversational agent like JavaPAL can be a good complement to the MOOC, especially for some types of learners, mainly those who are more accustomed to the use of mobile devices or for which the learning curve to use a conversational agent is not very high.

5. Conclusions

An important limitation in MOOCs is the lack of support to learners. This limitation is particularly critical in the case of MOOCs on engineering and computer sciences due to the intrinsic difficulty of the contents. Conversational agents may alleviate this problem of lack of support to learners, becoming learners' study partners. JavaPAL is a pioneering work on the use of voice-based conversational agents to support MOOC learners offering a quiz mode (JavaPAL asks the learner questions selected from the MOOC) and a review mode (JavaPAL provides definitions of the key concepts addressed in the MOOC as requested by the learner). This article has shed some light on the use of conversational agents through the example of JavaPAL, concluding that: 1) a conversational agent such as JavaPAL can provide a good usability to learners; 2) a conversational agent such as JavaPAL does not mean a worse learner's performance in terms of answering correctly questions from the MOOC; and 3) a conversational agent like JavaPAL can be a good complement to the MOOC for learners who are more used to using mobile devices.

Although the results obtained are encouraging, this research work is not without limitations, which should be addressed as future research. First, JavaPAL has been designed to support learners enrolled in a specific MOOC. More research should be done adapting this conversational agent to the contents (key concepts and questions) extracted from other MOOCs (not necessarily in the areas of engineering or computer sciences). Second, the number of users from which data was collected is 39 (for RQ1) and 15 (for RQ2 and RQ3). More research should be done with a higher sample of JavaPAL users, and particularly with a sample that contains learners who are indeed taking the MOOC for which JavaPAL provides support. After the prototyping phase the conversational agent is now ready to be offered to a large number of learners taking the MOOC. Third, the comparison between the conversational agent interface and the MOOC web interface was designed to be fair, although some existing questions from the MOOC had to be adapted (in the case of RQ2) to be used in JavaPAL (as show in Appendix A). It would be interesting to do the opposite and design questions in a MOOC directly to be used in a conversational agent and then transfer these same questions to the MOOC web interface.

Acknowledgment. This work was supported in part by the FEDER/Ministerio de Ciencia, Innovación y Universidades-Agencia Estatal de Investigación, through the Smartlet and H2O Learning projects under Grant TIN2017-85179-C3-1-R and PID2020-112584RB-C31, and in part by the Madrid Regional Government through the e-Madrid-CM Project under Grant S2018/TCS-4307 and under the Multiannual Agreement with UC3M in the line of Excellence of University Professors (EPUC3M21), and in the context of the V PRICIT (Regional Programme of Research and Technological Innovation), a project which is co-funded by the European Structural Funds (FSE and FEDER). Partial support has also been received from the European Commission through Erasmus+ Capacity Building in the Field of Higher Education projects, more specifically through projects LALA, InnovaT, and PROF-XXI (586120-EPP-1-2017-1-ES-EPPKA2-CBHE-JP), (598758-EPP-1-2018-1-AT-EPPKA2-CBHE-JP), (609767-EPP-1-2019-1-ES-EPPKA2-CBHE-JP). This publication reflects the views only of the authors and funders cannot be held responsible for any use which may be made of the information contained therein.

References

1. Shah, D., By the Numbers: MOOCs in 2019. (2019) [Online]. Available: https://www.classcentral.com/report/mooc-stats-2019 (current July 2020)
2. Cook, S., Bingham, P., Reid, S., Wang, X.: Going massive: Learner engagement in a MOOC environment. In Proceedings of the THETA 2015 - Create, Connect, Consume-Innovating today for tomorrow. Gold Coast, Queensland, Australia. (2015)
3. Alario-Hoyos, C., Pérez-Sanagustín, M., Delgado Kloos, C., Muñoz-Organero, M., Rodríguez-de-las-Heras, A.: Analysing the impact of built-in and external social tools in a MOOC on educational technologies. In Proceedings of the 8th European Conference on Technology Enhanced Learning (EC-TEL). Springer, Berlin, Heidelberg, Paphos, Cyprus, 5-18. (2013)
4. Yukselturk, E., Ozekes, S., Türel, Y. K.: Predicting dropout student: an application of data mining methods in an online education program. European Journal of Open, Distance and e-learning, Vol. 17, No. 1, 118-133. (2014)
5. Chyung, S. Y.: Systematic and systemic approaches to reducing attrition rates in online higher education. American Journal of Distance Education, Vol. 15, No. 3, 36-49. (2001)
6. Vitiello, M., Walk, S., Hernández, R., Helic, D., Gütl, C.: Classifying students to improve MOOC dropout rates. In Proceedings of the 2016 European MOOCs Stakeholders Summit (EMOOCs) Research Track. Graz, Austria, 501-508. (2016)
7. Vitiello, M., Walk, S., Helic, D., Chang, V., Guetl, C.: User Behavioral Patterns and Early Dropouts Detection: Improved Users Profiling through Analysis of Successive Offering of MOOC. Journal of Universal Computer Science, Vol. 24, No. 8, 1131-1150. (2018)
8. Dhorne, L., Deflandre, J. P., Bernaert, O., Bianchi, S., Thirouard, M.: Mentoring learners in MOOCs: A new way to improve completion rates. In Proceedings of the 2017 European MOOCs Stakeholders Summit (EMOOCs). Springer Cham, Madrid, Spain, 29-37. (2017)
9. Aguirre, C. C., Delgado-Kloos, C., Alario-Hoyos, C., Muñoz-Merino, P. J.: Supporting a MOOC through a Conversational Agent. Design of a First Prototype. In Proceedings of the 2018 International Symposium on Computers in Education (SIIE). IEEE, Jerez de la Frontera, Spain, 1-6. (2018)
10. Flecha, R.: Sharing Words: Theory and Practice of Dialogic Learning. Rowman & Littlefield, Lanham (2000)
11. Caballé, S., Conesa, J.: Conversational Agents in Support for Collaborative Learning in MOOCs: An Analytical Review. In International Conference on Intelligent Networking and Collaborative Systems (pp. 384-394). Springer, Cham. (2018)

12. Zadrozny, W., Budzikowska, M., Chai, J., Kambhatla, N., Levesque, S., Nicolov, N.: Natural language dialogue for personalized interaction. Communications of the ACM, Vol. 43, No. 8, 116-120. (2000)
13. Deepai.org. [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/conversational-agent (current July 2020)
14. Song, D., Oh, E. Y., Rice, M.: Interacting with a conversational agent system for educational purposes in online courses. In Proceedings of the 2017 10th International Conference on Human System Interactions (HSI). IEEE, Ulsan, South Korea, 78-82. (2017).
15. Chatbots magazine. [Online]. Available: https://chatbotsmagazine.com/chatbot-report-2019-global-trends-and-analysis-a487afec05b (current July 2020)
16. Weizenbaum, J.: ELIZA - A computer program for the study of natural language communication between man and machine. Communications of the ACM, Vol. 9, No. 1, 36-45. (1966)
17. AbuShawar, B., Atwell, E.: ALICE chatbot: Trials and outputs. Computación y Sistemas, Vol. 19, No. 4, 625-632. (2015)
18. Jia, J.: The study of the application of a keywords-based chatbot system on the teaching of foreign languages. (2003). [Online]. Available: https://arxiv.org/pdf/cs/0310018.pdf (current July 2020)
19. Edwin.ai. [Online]. Available: https://edwin.ai (current July 2020)
20. Knill, O., Carlsson, J., Chi, A., Lezama, M.: An artificial intelligence experiment in college math education. (2004). [Online]. Available: http://people.math.harvard.edu/~knill/preprints/sofia.pdf (current July 2020)
21. Niranjan, M., Saipreethy, M. S., Kumar, T. G.: An intelligent question answering conversational agent using Naïve Bayesian classifier. In Proceedings of the 2012 IEEE International Conference on Technology Enhanced Education (ICTEE). IEEE, Kerala, India, 1-5. (2012)
22. Kerry, A., Ellis, R., Bull, S.: Conversational agents in E-Learning. In Proceedings of the International Conference on Innovative Techniques and Applications of Artificial Intelligence. Springer, Cambridge, UK, 169-182. (2008)
23. Shaw, E., Johnson, W. L., Ganeshan, R.: Pedagogical agents on the web. In Proceedings of the third annual conference on Autonomous Agents (AGENTS'99), ACM, Seattle Washington USA, 283-290. (1999)
24. De Pietro, O., & Frontera, G.: Tutorbot: An Application AIML-Based for Web-Learning. Advanced Technology for Learning, Vol. 2, No. 1, 29-34. (2005)
25. Graesser, A. C., Person, N., Harter, D., and the Tutoring Research Group: Teaching tactics and Dialog in AutoTutor. International Journal of Artificial Intelligence in Education, Vol. 11, 1020-1029. (2000)
26. Demetriadis, S., Karakostas, A., Tsiatsos, T., Caballé, S., Dimitriadis, Y., Weinberger, A., ... Hodges, M.: Towards integrating conversational agents and learning analytics in MOOCs. In Proceedings of the International Conference on Emerging Internetworking, Data & Web Technologies. Springer, Cham, Tirana, Albania, 1061-1072. (2018)
27. Tomar, G. S., Sankaranarayanan, S., Rosé, C. P.: Intelligent conversational agents as facilitators and coordinators for group work in distributed learning environments (MOOCs). In Proceedings of the 2016 AAAI Spring Symposium Series, AAAI Press. (2016).
28. Creswell, J. W.: Educational research: Planning, conducting, and evaluating quantitative. Upper Saddle River, NJ: Prentice Hall. (2002)
29. Bangor, A., Kortum, P., Miller, J.: Determining what individual SUS scores mean: Adding an adjective rating scale. Journal of usability studies, Vol. 4, No. 3, 114-123. (2009)
30. Delgado Kloos, C. Alario-Hoyos, C., Muñoz-Merino, P. J., Catalán-Aguirre, C. & González-Castro, N.: Principles for the Design of an Educational Voice Assistant for Learning Java. In Proceedings of the International Conference on Sustainable ICT, Education, and Learning. Springer, Zanzibar, Tanzania, 99-106. (2019)

31. Landauer, T. K.: Behavioral research methods in human-computer interaction. In Handbook of human-computer interaction. North-Holland, 203-227. (1997)
32. Lewis, J. R., & Sauro, J.: The factor structure of the system usability scale. In International conference on human centered design. Springer, Berlin, Heidelberg, San Diego, California, USA, 94-103. (2009)
33. Sauro, J.: A practical guide to the system usability scale: Background, benchmarks & best practices. Measuring Usability LLC. (2011)

Appendix A

Questions asked during the interviews

1. What are the differences between using the conversational agent and the traditional web interface to answer multiple choice questions?
2. What are the advantages of using a conversational agent?
3. What are the disadvantages of using the conversational agent?
4. What are the advantages of using the web interface?
5. What are the disadvantages of using the web interface?
6. In which situations would you use the conversational agent instead of using the web interface?
7. In which situation would you use the web interface instead of using the conversational agent?
8. Can you compare the conversational agent and the web interface in terms of learnability?
9. Can you compare the conversational agent and the web interface in terms of usability?
10. Can you compare the conversational agent and the web interface in terms of utility?

Example questions from the MOOC and transformed for the conversational agent

Questions from MOOCs	**Questions from JavaPAL**
Users have to write the result in a box given a value of “n”. System.out.println(n%6)	What is the result of the operation 7 percentage 2?
Users have to write the result of the operation, taking into account the precedence of the operators.	What operation is computed first? Multiplication / Addition / Division
Select True or False: An array can be extended after it has been initialized.	Can an array be extended after it has been created? True/False
The term “application” is similar to… Program Algorithm	The term application in Java is similar to: Program Algorithm
It is possible to run a program multiple times simultaneously? False True	Is it possible to run a program several times simultaneously? True False
The processing unit (select the correct answer out of 4 possible answers): - Is the module that runs the programs	The processing unit is the module that executes the program: - True / False

Cristina Catalán Aguirre is currently Product Line Maintenance Manager in Cloud Core Data-Storage Manager design organization at Ericsson. She received a double master's degree in Telecommunication Engineering and Telematics Engineering from Universidad Carlos III de Madrid in 2019 and a bachelor's degree on Telecommunication Technologies Engineering from the Public University of Navarra in 2017. She worked as a research assistant at the Telematics Engineering Department of Universidad Carlos III de Madrid from 2017 to 2019.

Nuria González-Castro is currently Cloud Engineer in Openbank. She received a double master's degree in Telecommunication Engineering and Telematics Engineering from Universidad Carlos III de Madrid in 2020, and a bachelor's degree on Telecommunication Technologies Engineering from this same university in 2018. She worked as a research assistant at the Telematics Engineering Department of Universidad Carlos III de Madrid from 2017 to 2020.

Carlos Delgado Kloos received the PhD degree in Computer Science from the Technische Universitat Munchen and in Telecommunications Engineering from the Polytechnical University, Madrid. He is full professor of Telematics Engineering at the Universidad Carlos III de Madrid, where he is the director of the GAST research group, director of the UNESCO Chair on "Scalable Digital Education for All", and Vice President for Strategy and Digital Education. He is also the Coordinator of the eMadrid research network on Educational Technology in the Region of Madrid.

Carlos Alario-Hoyos is Visiting Associate Professorin the Department of Telematics Engineering at the Universidad Carlos III de Madrid. He received M.S. and PhD degrees in Information and Communication Technologies from the Universidad of Valladolid, Spain, in 2007 and 2012, respectively. His skills and experience include research and development in MOOCs, social networks, collaborative learning, or evaluation of learning experiences.

Pedro J. Muñoz-Merino is Associate Professor at the Universidad Carlos III de Madrid, where he got his PhD in Telematics Engineering in 2009. Pedro has received several awards for his research. He is author of more than 100 scientific publications and has participated in more than 30 research projects. His skills and experience include research and development in learning analytics, educational data mining, evaluation of learning experiences, and gamification, among others.

Received: July 31, 2020; Accepted: February 28, 2021.