**Experiment Setup**

In this experiment we had undergraduate and graduate students from McGill University (COMP-551) try out the system. The experiment was carried out in-class. Students were first given a 5 min introduction. This was followed by a 5 min demo of the system, highlighting how to write natural language answers and the concept tree building auxiliary task. Then the students spent 30-40min trying out the system and afterwards there was an open feedback session followed by a presentation about the system itself. The experiment was conducted on the evenings of February 6th and February 8th, 2019.

To handle disengaged students, students with less than 5 user utterances are excluded from this analysis. We also excluded 4 other students who were clearly abusing or playing around with the system instead of following the exercises given.

The goal of this experiment was to evaluate the four components: 1) automatically generated hints, 2) new student model, 3) concept tree visualization, 4) concept tree building, and 5) selection task.

We evaluated three variants of our system:

1. (Handicapped) System, where the student model is not utilized at all
2. (Handicapped) System, where the student model is not utilized by the policy for selecting next tutor engine states, and
3. (Full) Regular system, where the system uses all its modules including the student model

**Experiment Results**

All metrics are given in medians, unless otherwise indicated.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Metric (Avg)**  **System Variant** | **Sessions**  **(Users)** | **User Utterances** | **Exercise Problems Shown** | **Adjusted Session Duration (Min)** | **Average User Response Time (Min)** | **Average Exercise Duration (Min)** | **User Clicked**  **'I have a question' (Mean)** | **User Clicked 'I don't know' (Mean)** | **“Happy” Emoticon**  **(Mean)** | **“Surprised” Emoticon**  **(Mean)** | **“Angry” Emoticon**  **(Mean)** |
| Handicapped System, Excluding Student Model | 7 | **34** | **12** | 22.18 | 0.73 | 1.58 | 0.14 | 1.00 | **1.71** | 0.14 | **0.00** |
| Handicapped System, Excluding Student Model in Tutor Engine Policy Decisions | 14 | 30.5 | 11 | **23.61** | **1.06** | **1.89** | 0.79 | 0.93 | 0.43 | **0.21** | 0.00 |
| Full System | 15 | 31 | **12** | 22.27 | 0.83 | 1.71 | **0.93** | **2.47** | 0.53 | 0.07 | 0.33 |
| All Systems | 36 | 31 | 12 | 22.89 | 0.96 | 1.78 | 0.72 | 1.58 | 0.72 | 0.14 | 0.14 |

This table compares the current system variants with the systems from October, 2018.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metric (Avg)**  **System Variant** | **Sessions**  **(Users)** | **On a scale 1-5, how would you rate your overall learning experience with me? (Avg)** | **How often did the chat help you understand the material better? (Avg;**  **1=Never,**  **2=A few times, 3=Sometimes, 4=Many times, 5=Every time)** | **On a scale 1-5, how fun is the chat compared to learning by yourself? (Avg)** | **Ratio**  **Thumbs Up (Mean)**  **Over Thumbs Down (Mean)**  **/**  **Ratio**  **“Happy” Emoticon (Mean)**  **Over “Angry” Emoticon (Mean)** |
| All Systems, (October, 2018) | 30 | 3.32 | 3.26 | 3.50 | 2.27 |
| All Systems, (February, 2019) | 36 | **3.60** | **3.30** | **3.60** | **5.14** |

This table shows the impact of the concept tree visualization intervention. These statistics are based on manual annotations by Iulian Vlad Serban and exclude cases where the user had already given a “CORRECT” solution attempt but the system hadn’t detected

|  |  |
| --- | --- |
| Sessions (Users) | 36 |
| Number of times concept tree visualization was shown (including the times it was shown after the user clicked “I don’t know” or “I have a question”) | 17 (100%) |
| Number of times users asked a question immediately after concept tree visualization was shown | 8 (47.06%) |
| Number of times users improved their next solution attempt after concept tree visualization was shown | 10 (58.82%) |
| Number of times users gave a “CORRECT” next solution attempt after concept tree visualization was shown | 5 (29.41%) |
|  |  |
| Number of times preemptive doubt resolution with suggested questions was shown (including the times it was shown after the user clicked “I don’t know” or “I have a question”) | 63 (100% |
| Number of times users asked a question immediately after preemptive doubt resolution with suggested questions was shown | 53 (84.13%) |

This table shows the impact of the concept tree building task intervention. These statistics are based on manual annotations by Iulian Vlad Serban and exclude cases where the user had already given a “CORRECT” solution attempt but the system hadn’t detected it. Here, it was also observed that in several examples users were clearly re-using the concept names (node names and related concepts) in the concept trees when writing their answers.

|  |  |
| --- | --- |
| Sessions (Users) | 36 |
| Number of times concept tree building was shown (including the times it was shown after the user clicked “I don’t know”) | 25 (100%) |
| Number of times users improved their next solution attempt after concept tree building was shown | 14 (56.00%) |
| Number of times users gave a “CORRECT” next solution attempt after concept tree building was shown | 7 (28.00%) |

This table shows the impact of the selection task intervention. These statistics are based on manual annotations by Iulian Vlad Serban and exclude cases where the user had already given a “CORRECT” solution attempt but the system hadn’t detected it.

|  |  |
| --- | --- |
| Sessions (Users) | 36 |
| Number of times selection task was shown (including the times it was shown after the user clicked “I don’t know”) | 18 (100%) |
| Number of times users improved their next solution attempt after selection task was shown | 4 (22.22%) |
| Number of times users gave a “CORRECT” next solution attempt after selection task was shown | 2 (11.11%) |

This table shows the frequency with which users hover over definitions in the concept tree and selection tasks. These statistics are computed automatically and may include concept trees or selection tasks that were shown even if the user gave a “CORRECT” solution attempt before.

|  |  |
| --- | --- |
| Number of times users hovered over and were shown a a definition | 102 |
| Number of times concept tree visualization was shown (including the times it was shown after the user clicked “I don’t know” or “I have a question”) | 20 |
| Number of times concept tree building task was shown (including the times it was shown after the user clicked “I don’t know”) | 33 |
| Number of times selection task was shown (including the times it was shown after the user clicked “I don’t know”) | 19 |
| Average number of concept definitions shown for each concept tree visualization, concept tree building or selection task | 1.42 |

This table shows the impact of the text-based hints intervention. These statistics are based on manual annotations by Iulian Vlad Serban and exclude cases where the user had already given a “CORRECT” solution attempt but the system hadn’t detected it.

|  |  |  |
| --- | --- | --- |
|  | **Human-Generated Hints** | **Machine-Generated Hints** |
| Sessions (Users) | 36 | 36 |
| Number of times text-based hint was shown (including the times it was shown after the user clicked “I don’t know”) | 30 (100%) | 19 (100%) |
| Number of times users improved their next solution attempt after hint was shown | 8 (26.67%) | 8 (42.11%) |
| Number of times users gave a “CORRECT” next solution attempt after hint was shown | 5 (16.67%) | 6 (31.58%) |

Although the machine-generated hints appear to be performing on par or better than the human-generated hints based on these statistics, it should also be noted that several nonsensical machine-generated hints were observed such as:

*EXERCISE: What is the difference between overfitting and underfitting? Explain.*

*HINT: Let me give you a little nudge in the right direction. Recall that setting it to zero and solving via calculus.*

*EXERCISE: What is the difference between regression and classification?*

*HINT: Let me give you a hint. Note that the given values might belong to.*

These were the student answers, when after 20 minutes Lila asked them: “On which of these elements is it most important for me to improve?”

|  |  |
| --- | --- |
| **Element** | **Percentage (Users)** |
| Ask me better questions. | 22.72% (5) |
| Give me better feedback and hints | **36.36% (8)** |
| Get better at answering my questions. | 9.09% (2) |
| Get better at identifying if my answer is right or wrong | **27.27% (6)** |
| Get better at entertaining me during the chat. | 4.55% (1) |

These were the student answers, when after 20 minutes Lila asked them: “How often did you find the concept trees useful?”

|  |  |
| --- | --- |
| **Element** | **Percentage (Users)** |
| Never | 9.09% (2) |
| A few times | 13.64% (3) |
| Sometimes | 18.18% (4) |
| Many times | **27.27% (6)** |
| Every time | **31.82% (7)** |

**These were the student answers, when after 20 minutes Lila asked them: “What excited you most about the chat?”**

*- The different types of questions*

*- The concept tree part.*

*- good*

*- Responsiveness*

*- Seems to have a lot of interactive tools to teach*

*- drag n drop trees*

*- Getting more practice questions was nice*

*- it's interactive*

*- It contains informative contents*

*- the step backs and clear explanatory diagrams*

*- That's it's interactive*

*- The fact that I knew there is someone who knows the answer and can help me*

*- Its smartness and interactiveness. Hell of a job! Congrats*

*- Some helpful test on my basic knowledge of machine learning.*

*- You can understand me.*

*- The responses are quite natural*

*- the hints*

*- You're good at understanding what I'm saying!*

*- The interaction*

*- it's different*

*- Interaction*

*- How smart it is*

*- It response well.*

**These were the student answers, when after 20 minutes Lila asked them: “What frustrated you most about the chat?”**

*- Sometimes my answer would be really close to the answer, but needed to be worded differently, and I sometimes wouldn't know what extra information is needed*

*- Sometimes the question is not so clear.*

*- nothing*

*- Nothing in particular*

*- Very guided, unidirectional approach, not much freedom*

*- sometimes answers required higher degree of specificity than the question conveyed*

*- The syntax was sometimes unclear*

*- sometimes it doesn't understand the real meaning of the sentences*

*- Nothing*

*- its hard to know what exactly to communicate because theres so many ways to say the same thing*

*- sometimes it was hard to understand what it wanted of the answer*

*- you do not understand lots of things :) (no offense ;))*

*- I believe sometimes it didn't fully understand my answers*

*- Actually, sometimes I just guess and choose from the choices and want some feedbacks but nothing appeared.*

*- Maybe have questions come in more formats*

*- I wish I could answer questions about a wider variety of topics*

*- the notations are kind of messy. it would be better if you could define what these subscript mean*

*- Nothing really*

*- Nothing*

*- looks for particular keywords*

*- Not having the perfect answers*

*- equations can be not clear*

*- They cannot understand some daily words.*

**Discussion**

Student Model and Tutor Engine Policy

The A/B test shows that the differences between system variants are relatively small in general.

The A/B test shows that utilizing the student model to improve solution verification increases the overall user experience (see first table rows "Handicapped System, Excluding Student Model" and "Handicapped System, Excluding Student Model in Tutor Engine Policy Decisions") according to the overall session duration ("Adjusted Session Duration"), user response time ("Average User Response Time") and average exercise duration ("Average Exercise Duration"). These results suggests that students are willing to spend more time with this system and dive deeper into each exercise. The same conclusion is supported by the number of "I have a question" clicks given for each system variant. The only metric contradicting this assumption is the number of happy emoticons clicked ("Happy" Emoticon), which is higher for the system not utilizing the student model. However, the difference is less than a dozen clicks and in past experiments we have found a high variance in similar metrics (thumbs up and thumbs down metrics).

The A/B test also shows that additionally utilizing the student to select appropriate interventions (see first table rows "Handicapped System, Excluding Student Model in Tutor Engine Policy Decisions" and "Full System") has no significant impact on performance. The reason for this may be due to 1) the limited data available to train the student model (for example, the student model has been trained on very few examples of concept tree, selection task and missing concept interventions), 2) the changing system interventions (for example, concept tree and missing concept interventions have been significantly improved since the last experiment, and this may not be reflected in the student model), and 3) the criteria (reward function) used by by the student model to select an appropriate intervention, which is to select the intervention with maximum probability of the student answering the exercise correctly.

The A/B test strongly suggests that main role of the student model in the current system (with the its current modules, feedback interventions and limited training data) should be to aid the solution verification model. As discussed above, it appears to already have a positive impact here. The need to improve the solution verification model accuracy has also previously been established throughout the past 6 experiments with university students. In other words, the student model should only be used in conjunction with the solution verification model to form an ensemble model. Here, the student model is optimized only to increase the overall solution verification accuracy of the ensemble model.

In this case, we should learn a new tutor engine policy separate from the student model. This policy should be based on a few simple features such as 1) length/complexity of previous student answer (if any), 2) time taken to write previous student answer (if any), 3) accuracy of previous student answer (if any), and 4) number of “I don’t know” clicks, 5) number of “I have a question” clicks, and 6) number of questions asked. The policy should then output a value for every possible intervention. The policy can be learned from manually annotated data (e.g. 1000 annotated examples).

Concept Trees

The experiment shows that the concept tree visualization and concept tree building task interventions are useful interventions. In over half of the cases, these two interventions help the students to improve their answers. In 25%-30% of the cases, the students give a correct answer immediately after each intervention. In addition over half of the students said they found the concept trees useful either “Many times” or “Every time”. Due to the generality of this approach and other downstream use cases (e.g. selection task module, competing facts module, solution verification input features etc.), we should continue our efforts on building up a pipeline to extract concept tree.

The qualitative analysis shows that students tend to re-use the terms in the concept tree (including the relevant concept nodes). This is supported by the recorded hovering events, which shows that users on average hover over 1.42 definitions per concept tree. Therefore, it will be important that the concept tree extraction algorithm also extracts 1) accurate definitions and 2) accurate related concepts.

Since in nearly half of the cases students also asked follow up questions, it is important to keep the suggested questions as part of the concept tree visualization. It is also likely to be beneficial to generate more complex questions beyond the existing, simplistic what-questions (for example, questions like “What are the practical uses of linear regression?” and “How are linear regression models often fitted?”).

Selection Task

The selection task might be the poorest performing auxiliary task right now. It only helped improved student answers in 22.22% of the cases where it was shown. In previous experiments, students have also complained about the ambiguity of the task.

However, we have also not spent much (if any) time to improve this task over the last few months. It is likely that by improving the description of the task and the quality of the concepts shown, the selection task may increase its impact significantly. The impact of the intervention may also be increased by showing an example of how to do the task or by asking students to start by selecting the most relevant item first.

Human-Generated vs Machine-Generated Hints

The text-based hints generated by humans or by a machine automatically perform reasonably well. In previous experiments, we found that human-generated hints helped users to improve their answer by on average 10-30% points (i.e. the probability of being “CORRECT” after a hint increased by 10-30% points). The table above suggests that the hints in this experiment help improve the student answers 20% - 50% of the time and that 10% - 30% of the time the student’s next solution attempt is “CORRECT”.

When comparing the human-generated hints to the machine-generated hints, the table above shows that the machine-generated hints perform better. This may be an artefact of the fact that the machine-generated hints are created based on the reference solutions and therefore may reveal more information to the students. These numbers should therefore be interpreted with a grain of salt. If we stay on the conservative side, we may conclude that the machine-generated hints are overall comparable to the human-generated hints. This is still good news, because it means that we can automate one of the most important parts of the content creation pipeline.

However, there is still much room for improvement. The two major improvements to focus on are 1) automatically generating higher quality hints (more fluent, more relevant hints), 2) personalizing the hints shown based on the student’s last solution attempt, and 3) personalizing the hints shown based on the overall student profile.