A Novel Approach to Automated Content Generation for Education using GPT-3

Anonymous EACL submission

Abstract

The primary objective of this research is to investigate automatic content generation in an educational context. In an era characterized by an unprecedented influx of information, the conventional methods of content creation for classroom instruction have been rendered increasingly inadequate, thus the motivation behind this research is to aid teachers in generating content for educational use such that they won't need to expend much time and energy as with traditional methods.

Modern methods of generating content for the classroom are sought after due to the benefits when compared with more traditional methods. One example of this is a case study carried out amongst 48 college students where a positive effect occurred in the students' learning outcomes when they used computer-generated questions.

With automated content generation being the primary focus of this research, this research heavily relies on and investigates Natural Language Processing (NLP) techniques and technologies. Thus we delve into how automated content generation for previous systems was carried out along with Large Language Models (LLMs)

Our methodology relies on making use of the GPT model, GPT-3, the proposed system performs various NLP tasks such as Summarization and Information Retrieval (IR) along with prompt engineering to generate content within an educational context and empower educators when it comes to generating content. The system accepts inputs from the user that may be plain text, a YouTube video or a PDF and then generates content, such as a worksheet with questions in return by interfacing with and using GPT-3 to generate the content.

One also must keep in mind that such a system raises ethical qualms, particularly regarding data privacy and bias. Algorithmic bias is a commonly known issue within the field of NLP, as bias often arises from biased training data and algorithms. This bias can be harmful as it can directly affect the learning outcomes of certain groups of students. Furthermore, as such a system may collect learner data, data privacy comes into question, particularly who or what has access to this data and how it is used. A limitation of the currently proposed system is that as it uses GPT-3 as a backend, it will incorporate the same bias as GPT-3. The system however does not pose a data privacy risk as no sensitive or personal information is asked for, and the given inputs are only retained up until the corresponding output is generated.

In conclusion, this research focuses on the integration of computational linguistics within the field of education through the integration of GPT-3 with the application of automated content generation. The results of this study show a positive trend as 94% of the respondents said that the system generated relevant content while 85% of respondents said that they would adopt such a system.

This work raises the question of how NLP can be utilised more effectively within the field of education. Furthermore, this system, while currently aimed at primary and secondary level students at a general level, in future work it can potentially be adapted for particular grade levels and particular topics by fine-tuning the model.

1 Introduction

As the demand for top-notch education rises, the interest in educational tools such as Intelligent Tutoring Systems (ITS) has also risen. Tools such as the one mentioned mark a significant progression in the field of educational technology.

An ITS is a tech-driven system that is primarily focused on offering students an individual and carefully curated learning experience. This approach to education is not only innovative but also promising as it is set to surpass the more traditional methods of teaching. What sets ITSs apart in education is their ability to track a student's progress and understanding throughout their educational journey. (Graesser et al., 2018)

This active interplay between technology and pedagogy ensures that any barriers to a student's progress are quickly identified and addressed, creating a more supportive and effective learning environment. However, despite the undeniable benefits of ITS, it's crucial to recognize a practical consideration.

As educators traverse the complex landscape of the learning process, they inevitably reach a point where they must create custom content that aligns perfectly with their classroom's specific goals and student demographics. This customisation while crucial for delivering a personalised learning experience, requires a significant commitment of time and effort that could be better spent on other educational tasks. (Nkambou et al., 2010)

The proposed system interfaces with the GPT-3 model Curie variant to perform various tasks such as generating lesson plans and creating questions for students. For example, the system can assess a reading comprehension passage and generate a list of related questions. This system has the potential to revolutionize education by changing how content is created, facilitating content distribution, and introducing the possibility of educators collaborating and sharing resources through this system.

In this study, we explore the potential of Natural Language Processing NLP models and techniques to assist educators in content generation. Our hypothesis centres on the capabilities of GPT-3, a model that was considered state-of-the-art at the time of its introduction, and its potential to significantly contribute to this area.

1.1 Objectives

This research seeks to explore the necessary methods and technologies for constructing a system capable of generating educational content from an NLP prompt and to understand the potential impact of such a system on networked learning.

The following objectives have been outlined:

- Examine cutting-edge NLP models and techniques that are pertinent to the development of the proposed system, with a particular focus on the GPT series of models.
- Identify suitable NLP models and techniques for a system that can produce classroom content from a natural language prompt.

1.2 Motivation

The inspiration of this research is the exciting possibility of automated content creation within an ITS, this would significantly reduce the time and effort educators need to invest in content generation.

The primary aim of this project is to make use of automation to streamline and improve the creation of educational materials, thereby addressing several key challenges in modern education. Fundamentally, automated content production signifies a shift in how we perceive and implement personalized learning environments.

In conventional educational scenarios, customizing instruction to suit each student's individual needs and learning styles can be labour-intensive and time-consuming. However, the introduction of automated content creation opens up the potential to transform this aspect of education. One of the main benefits of automated content production within ITS is its ability to greatly enhance the scalability of customized learning environments.

Essentially, it enables educators and ITS developers to efficiently produce a wide variety of tailored educational materials without the time and resource constraints typically associated with manual content creation. This scalability is crucial, especially in educational settings where a diverse group of students with varying learning needs require access to high-quality instruction.

2 Literature Review

2.1 Intelligent Tutoring Systems

The primary objective of any ITS is to deliver a personalised one-on-one education through AI and Machine Learning, enhancing the learning process through guidance and feedback which are instant, along with the analysis of learner behaviour to adapt to the learner's needs and preferences.

An ITS is composed of key elements that work together to provide personalized learning experiences:

- Student Model: The student model is a representation of the learner's performance, knowledge and preferences. As the learner progresses, the student model evolves, thus enabling personalised feedback. The student model is updated through sources such as assessments and interactions throughout the ITS. (Chrysafiadi and Virvou, 2013)

- Tutoring Strategy: Perhaps one of the most prominent aspects of an ITS is the tutoring strategy. This component has a focus on modelling the domain knowledge and providing feedback and guidance to the student based on the student model. (Nwana, 1990)

- Content Library: The content library of an ITS can include multimedia resources such as text, images, and videos. These resources enable flexible and personalized learning experiences, adapting to changing educational demands and staying current with advancements. (Nwana, 1990)

- User Interface (UI): The interaction between the learner and the system is carried out via the UI. It facilitates learning and engagement by offering progress tracking, clear instructions and adaptive content. (Lopes et al., 2019)

The components of an ITS work in tandem to provide the learner with an adaptive, personalised education through the understanding of the learner's unique needs and progress.

2.1.1 Benefits & Challenges

ITSs strive to deliver a personalised learning experience, adjusting to the unique needs of each student through the analysis of interactions via machine learning. Providing immediate feedback, is beneficial, especially for more complex subjects.

ITS also adapt their teaching methods to align with students' learning styles, be it visual or auditory. (Graesser et al., 2018) These systems offer instant access to current educational resources, encompassing libraries, simulations, and games. Educators gain from automated grading and the capability to pinpoint students who are struggling for timely intervention.

Despite their benefits, the development and upkeep of ITS can be expensive, and evaluating their effectiveness poses a challenge. Nevertheless, they broaden educational access worldwide and cater to a diverse range of learning demographics.

2.1.2 Automated Content Generation Within ITS

Automated content creation has the benefit of quickly producing a large amount of information, which is particularly useful in fields such as science and technology. Additionally, it can tailor content to individual students by analyzing their learning preferences, progress, and performance data.

An example of this is MathBot, a conversational chatbot that provides students with feedback. The system uses a conversational graph to generate questions and guide the conversation. When it detects a flaw in the learner's logic, it reviews earlier concepts. (Grossman et al., 2019)

2.2 Large Language Models

Large Language Models known as LLMs are capable of understanding natural language and producing text that appears to be written by a human. These LLMs utilize deep neural networks, a type of machine learning model that excels at identifying complex patterns in data. During the training phase, LLMs are exposed to vast amounts of text data, including novels, news articles, and web pages. (Cheng et al., 2023) They learn to predict the next word in a sentence based on the preceding words, thereby developing a deep understanding of the relationships between words and their context.

Deep neural networks form the basis of large language models. They process input data through interconnected nodes in layers, producing a coherent sentence. Training these models involves adjusting node weights to minimize the difference between predicted and actual output, using a set of ideal word sequences for comparison. (Schwenk and Gauvain, 2005)

2.2.1 Transformers

Transformers, initially released by (Vaswani et al., 2017), is an LLM model which had an everlasting impact on the field of NLP. Before the introduction of this model, Recurrent Neural Networks, Long Short-Term Memory and Gated Recurrent Neural Networks were established as the state-of-the-art approaches to language modelling, a position that has since been delegated to transformers.

While the attention mechanism had been utilised previously, the Transformer was the first to rely entirely on the attention mechanism to learn patterns between the model's input and the output which is of dimension 512.

Architecture

The Transformer architecture comprises an encoder-decoder structure. The encoder translates the input (symbolic representation) into a sequence of continuous representations. Given this continuous representation, the decoder will generate an output sequence which is of a symbolic representation. Furthermore, the Transformer is an auto-regressive model, meaning that it takes the output it has just generated as additional input when generating the next token in the output.

Encoder & Decoder

The Encoder consists of six identical layers, each containing two sub-layers. The first sub-layer utilizes a multi-head self-attention mechanism, and the second one is a position-wise fully connected feed-forward network. Each sub-layer is surrounded by a residual connection, followed by layer normalization.(Vaswani et al., 2017)

The Decoder, like the Encoder, is composed of six identical layers. However, each layer in the Decoder has three sub-layers. The first two are identical to those in the Encoder, and the third one performs multi-head attention over the Encoder's output. Each sub-layer in the Decoder also has a residual connection followed by layer normalization. The self-attention sub-layer in the Decoder is modified to prevent positions from attending to subsequent positions, ensuring that predictions at position i depend only on known outputs at positions less than i.(Vaswani et al., 2017)

Attention

An attention function can be understood as a method that maps a query and a collection of key-value pairs to an output. All of these elements - the keys, values, query, and output - are represented as vectors. The output is generated by transforming the sum of the values. Each value is assigned a weight, which is determined by a function that takes the query and the corresponding key as inputs.(Vaswani et al., 2017)

Positional Encoding

Transformers lack recurrence or convolution, which means they need additional information to utilize the order of the input. This is where positional encoding comes into play. It provides information about the relative and absolute positions of the tokens (words) in the sequence. These encodings are added to the input embeddings at the base of the encoder and decoder stacks. Since the encodings and embeddings share the same dimensions, they can be summed together. (Vaswani et al., 2017)

Conclusion

By making use of the concepts discussed, the transformer model sets itself apart from other machine-learning models. In recent years, development within the NLP field has only cemented the place of transformers as a state-of-the-art model.

2.2.2 GPT-3

GPT-3 is a 175-billion-parameter auto-regressive language model that was built as an improvement upon the previously existing GPT-2 model, when introduced GPT-3 was notable for its strong performance on tasks such as machine translation and question-answering amongst others. (Brown et al., 2020)

When introduced GPT-3 was evaluated upon three conditions:

- Zero-Shot Learning: Zero-Shot learning is when the model predicts the given answer from a description of the given tasks.
- One-Shot Learning: The model is given the task description, along with a singular example of the given task.
- Few-Shot Learning: Few-Shot learning is when the model is given a task description along with some examples of the given task.

When evaluated, GPT-3 achieved promising results in the zero-shot and one-shot learning settings. Achieving an 81.5 F1 score on CoQA in a zero-shot setting, 84.0 F1 on CoQA in the one-shot setting, and 85.0 on a few-shot setting.

GPT-3 demonstrates proficiency in one-shot and few-shot tasks that require immediate reasoning or quick adaptation. This includes tasks like unscrambling words, performing arithmetic calculations, and using new words in a sentence after only a single exposure to their definitions. (Brown et al., 2020)

2.2.3 T5

T5 is an open-source model released by Google trained on a text-to-text framework. The text-to-text framework enables T5 to perform a multitude of tasks, with the task being performed dependent on the given prompt. (Raffel et al., 2019)

The T5 model, grounded in the transformer architecture, leverages the self-attention mechanism. This mechanism enables the model to focus on various parts of its input sequence during both encoding and decoding processes. Uniquely, T5 adopts a text-to-text framework where both the input and output are sequences of text. This design broadens the scope of NLP tasks that T5 can be trained for, encompassing areas such as question answering and text classification, among others.

T5, trained on a diverse text corpus, underwent two training stages. Initially, it predicted masked words in sentences in an unsupervised manner. Then, it was fine-tuned on specific NLP tasks. This pre-training allowed T5 to develop a generalized understanding of language, enhancing its task-specific performance.

2.3 Reinforcement Learning

When training a machine learning model, one might use reinforcement learning (RL), RL is a training process where the model learns based on its interaction with the environment. The model's objective is to maximise a variable called its reward. (Li, 2022)

We define a set of parameters to be the weights and biases of the model to parameterise a policy. Mathematically speaking, in RL we seek to maximise the reward by following this parameterised policy. Typically we need to create a reward function. (Li, 2022)

The model then takes a set of actions called state and action pairs, with the total reward being the outcome of these steps. A common approach to finding the set of parameters is to use the Gradient Ascent.

We use RL when modelling language because of the problem of how we define an acceptable answer from a machine learning model. Furthermore, we want the models to produce not only high-quality answers but answers that are free from bias. As a loss function which captures these attributes is difficult to design, human feedback can be opted for as a measurement of the performance of a model, this is often called Reinforcement Learning from Human Feedback (RLHF)

2.3.1 ChatGPT Training Process

In the initial step of training ChatGPT, there was a need to collect data and train a supervised policy first. Human trainers played out conversations where they took both the role of the user, as well as that of the AI assistant. Then a pre-trained model is fine-tuned on the dataset curated by the human trainers along with the old dataset. The given prompts are diverse and include a variety of tasks, including but not limited to question answering, dialogue, summarisation, natural language generation, etc... (Phan, 2020)

The next step is to obtain a model that takes an input pair comprising of a prompt and a text and returns a scalar reward which should represent the humans' preference, this model is the reward function approximated. The fine-tuned model from the initial step is tasked with generating k text samples to an input prompt. Then a human labeller will organise the generated samples in order from best to worst. Since humans might rate a result differently from one another, the reward model is trained on all the human-labeled results as a single batch for each prompt. This is computationally efficient and avoids over-fitting the model.

The loss function is then designed according to the reward model for a prompt x and the corresponding output y. If the reward for the completion being looked at is higher than the reward of the other completion being considered, then the loss is small. The supervised fine-tuned model with the final unembedding layer replaced takes a prompt and a response and outputs the reward thus it can be trained as a reward model.

The objective, which is determined using the reward model from the second step and the fine-tuned model from the first step, is defined with several components. These include the Kullback-Leibler reward coefficient, which manages the intensity of the Kullback-Leibler penalty, and a pretraining loss coefficient that oversees the pretraining gradients and the Kullback-Leibler penalty.

The model is then updated in several iterations using Gradient Ascent, with steps 2 and 3 iterated continuously. The resulting model (called InstructGPT) was then compared to GPT-3, its outputs were given higher scores than that of GPT-3 while having fewer parameters by a magnitude of 100. This model also showed an improvement in the truthfulness of its outputs and an improvement in toxicity.

3 System Requirements

As established previously, the primary aim of this research is to identify the most effective approach to designing and creating a system that can supply educators with educational materials. The fundamental premise of this system is to simplify and expedite the process of generating educational content for educators.

To fulfil this premise, the system needs to be simple and user-friendly, allowing teachers to access the information they need while eliminating unnecessary complexities. The system needs to be intuitive and dependable.

It's crucial to emphasize that such a system is not intended to replace educators, their expertise, or their judgment. Instead, this tool is designed to provide a supplementary resource that enhances and complements the educator's work in the classroom. It allows them to devote more time and energy inside the classroom while also enabling the preparation and organization of study materials more efficiently and with less effort.

Therefore, the system has three primary functions:

- Input through the form of text is received.
- Analysis of the given input and processing of the given information.
- After the input has been analyzed and processed, the system should provide a response that is both accurate and appropriate.

4 Methodology

The system being proposed in this research utilises prompt engineering in conjunction with an LLM. Prompt engineering allows us to guide the model towards our desired output by refining the inputted prompt and being explicit about what is required of the model. This facilitates the LLM to generate outputs which are contextually appropriate and accurate. The LLM being used in this case is GPT-3 which has already been trained on a vast dataset, giving the LLM a good comprehension of language and allowing it to generate accurate and appropriate outputs. The system has various functions designed to facilitate an educator's workload.

The UI comprises a grid of functions each being accompanied by an interface designed to be user-intuitive, thus enabling the the interaction between the user and the system. The functions included include the generation of questions, ideas and the summarisation of text amongst other functions.

4.1 Prompt Engineering

Once a function has been selected, the system will gather the input from the user (Figure 1) and start prompt engineering to carry out the desired function. Prompt engineering is handled by the back-end part of the system thus this is not visible to the user. Thus, before being sent to GPT-3 the prompt is modified in some way, typically by appending text before or after the given input. (White et al., 2023)

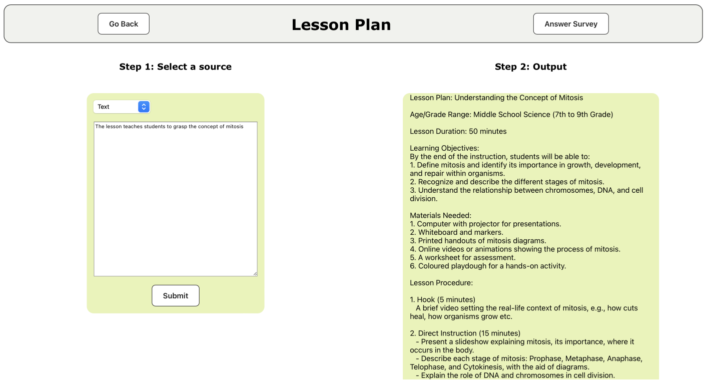

Figure 1: Collecting The Input

The prompts are similar to the zero-shot learning prompt. Where the textual prompt was written beforehand and then the task description would be included afterwards, an example of this is when making use of the text correction function, the engineered text would be "Correct the following text: This food are good. [NEWLINE] Corrected text:" and the model would then generate a correction of the given text.

4.2 Training The Model

The training of a machine learning model is the process of supplying the model with data to learn from, this typically includes some form of input data and some corresponding output data. This allows the model to learn from the patterns present within the data and make predictions on new unseen inputs.

In our specific application, the GPT-3 model implemented was fine-tuned via a synthetic dataset to improve response quality. A synthetic dataset is a dataset which has been computer-generated, versus a dataset that has been assembled through traditional means such as manual data collection. Synthetic data has its advantages such as being able to generate as much data as you need in little time.

This synthetic dataset thus manages to cover a broad range of scenarios given the functions the model was given. The synthetic dataset was created through GPT-3 itself by making use of randomly generated inputs and examples as the input for training and then using the model's output as the output value for training. This approach was chosen over real-world data due to the ease of collecting AI-generated responses rather than manually crafting both the input and output values for training the model.

The synthetic dataset comprised of input-output pairs such as "Write a poem about: apples" with a corresponding output pair. These were then sent to the model for fine-tuning. Fine-tuning is the further training of a pre-trained model which is used as a base model which is trained on further data.

4.3 System Architecture

For an efficient system, it was decided to split the system into two halves, the back-end and the front-end. This division aimed at enhancing performance and simplifying development by assigning code that is responsible for interfacing with the user to the front end while assigning code that interfaces with GPT-3 to the back end.

The back end is responsible for server-side tasks, particularly communication with the LLM. On the other hand, the front end handles user interface interactions, including web pages and visual elements. This division allows for a compartmentalized approach to application development, ensuring that changes in one area don't disrupt the other, thereby facilitating long-term maintenance.

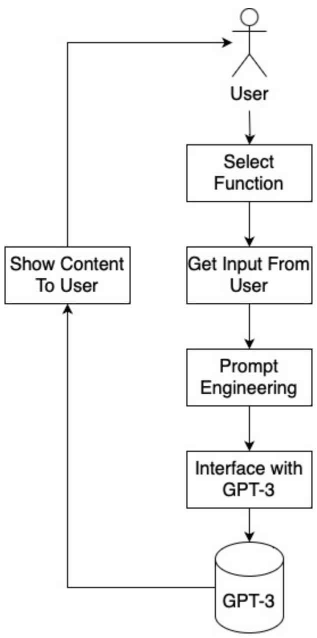

Figure 2: System Architecture

The system takes advantage of the React framework, renowned for its modular architecture, which simplifies complex user interface development. Using a component-based approach, features are built, tested, and individually integrated for enhanced reliability. Additionally, the system employs a server that manages both front-end and back-end tasks, improving efficiency and responsiveness while streamlining management, deployment, and maintenance processes.

5 Evaluation

To assess the system and methodology of this study, various aspects were identified and analysed. A questionnaire was subsequently distributed to gauge real-world user experiences with these system aspects.

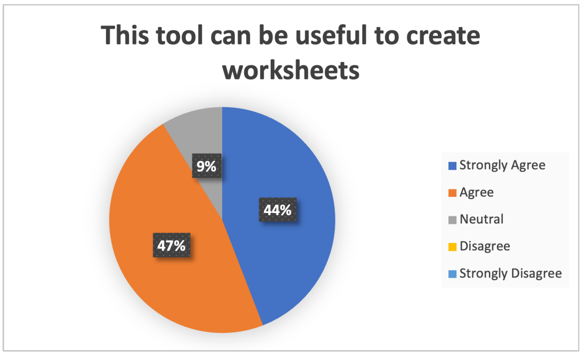

Figure 3: This tool can be useful to create worksheets

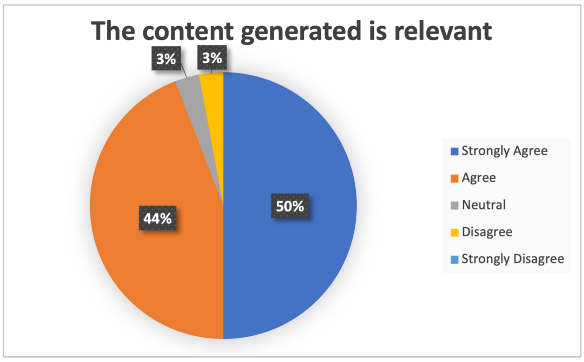

Figure 4: The content generated is relevant

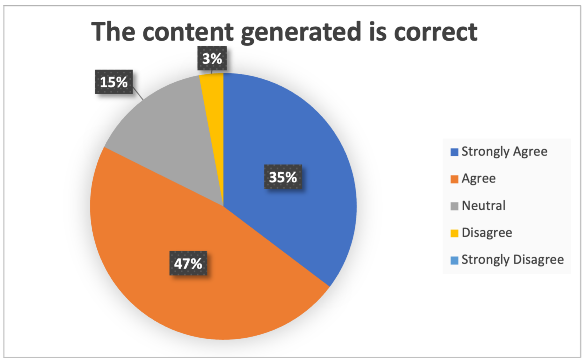

Figure 5: The content generated is correct

The initial three questions (Figures 3, 4, 5) in this study were meticulously selected to evaluate the system's proficiency in generating accurate and relevant content. The fourth question was designed to gauge the potential acceptance and adoption of such a system in real-world scenarios. The results from the first three questions indicate a commendable performance by the system in content generation.

Furthermore, the question "I would use this tool in my classroom" had a 47% response rate (Strongly Agree) and 38% (Agree) showing an 85% rate of system adoption if deployed. (Figure 6)

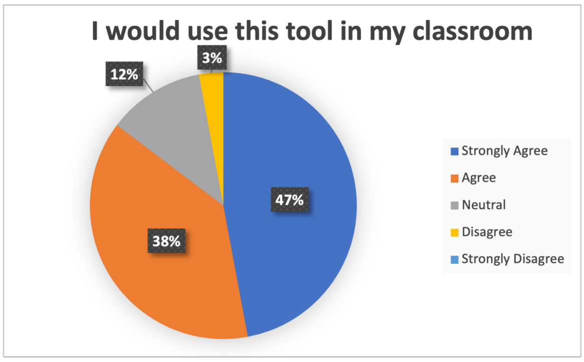

Figure 6: I would use this tool in my classroom

The response to the fourth question suggests a promising adoption rate for the system upon deployment. The Chi-Squared Test was run on the first three questions against the fourth question. The null hypothesis for this test is that the responses for the first three questions do not correlate to the respondent's willingness to adopt the system in the future, while the alternate hypothesis is that the responses do correlate to the willingness to adopt the system. When run, all tests scored a p value less than 0.05, hence we reject the null hypothesis and accept the alternate hypothesis, meaning that the positive responses given to the first three questions indicate a positive adoption rate.

The fifth question in this study was designed to ascertain whether the system could enable students to engage in a more hands-on approach to their learning. This question was posed to assess the system's potential to assist not only educators but also students directly. The responses were somewhat divided, with 64% of respondents expressing optimism that the system would indeed facilitate such an approach. However, 36% of respondents did not share this view. The reasons for these negative responses varied, ranging from concerns that the system might end up doing the students' work for them, to the belief that a physical approach is necessary for effective learning.

To evaluate the potential benefits of the system when utilized directly by students, respondents were queried about their perceptions of the system's impact on learning outcomes. A substantial 79% of respondents expressed a positive outlook, suggesting that the system could enhance learning outcomes. Among the reasons cited for this positive response was the belief that the system would foster greater student engagement.

6 Conclusion

LLMs hold tremendous promise for enhancing language interaction, including creative writing and natural language processing. However, their application brings up important ethical issues such as data privacy and bias. It's crucial to establish protective measures and ethical standards, as any bias in the LLM could result in unjust outcomes for certain student groups. (Weidinger et al., 2021; Baker and Hawn, 2021; Zhang et al., 2023) Additionally, it's essential to address concerns about the type of data collected when these systems are used. In the future, these concerns could be mitigated by implementing an algorithm on the front end such that the data being sent to the server is stripped of any personal/sensitive data, additionally biased outputs can be avoided by implementing checks which check outputs for bias.

Despite these challenges, LLMs have enormous potential in the field of NLP. They are transforming language interaction, enhancing communication, and pushing the boundaries of research across various fields.

In summary, the findings presented in the evaluation demonstrate that the proposed system exhibits a robust capability for content generation within an educational context and holds potential for further expansion to interact directly with students. This underscores the promising potential of automated content generation in educational settings when integrated with LLMs.

Future enhancements to this system would involve further exploration of how to improve automated content generation, integration of GPT-4, expansion of the system to allow direct use by students and the fine-tuning of the model used to generate subject-specific content at a higher quality. The system could also evolve to generate not only text based content, but multimedia content.

Limitations

While the system has shown to have good results when evaluated, the system still has it's limitations which will be discussed in this section.

With regards to privacy and bias, while the system does not collect any private or sensitive data from users, the inputted data is only processed in the prompt engineering phase after which it is sent to the LLM and no record of the input is kept, however it has been built upon GPT-3 thus it will exhibit the same limitations. Particularly:

- Data Bias: As GPT-3 is trained upon a large collection of data, it is not possible to review all of the training data for bias. Thus, the training data has biases that reflect society's. Due to this bias is captured in the model.
- Contextual Understanding: As GPT-3 is limited to its training training data, it does not have access to real-world information. Thus on occasion, it's outputs could be incorrect.
- Dependency on prompts: The GPT-3 model has a dependency on high-quality prompts to generate high-quality output, thus it was important to use prompt engineering within this research.

Concerns about scalability come to mind as such a system requires privacy and security checks along with quality assurance and maintenance, all of which get progressively more difficult when the system is scaled up.

References

Ryan S Baker and Aaron Hawn. 2021. Algorithmic bias in education. *International Journal of Artificial Intelligence in Education*, pages 1–41.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Daixuan Cheng, Shaohan Huang, and Furu Wei. 2023. Adapting large language models via reading comprehension. *arXiv preprint arXiv:2309.09530*.

Konstantina Chrysafiadi and Maria Virvou. 2013. Student modeling approaches: A literature review for the last decade. *Expert Systems with Applications*, 40(11):4715–4729.

Arthur C Graesser, Xiangen Hu, and Robert Sottilare. 2018. Intelligent tutoring systems. In *International handbook of the learning sciences*, pages 246–255. Routledge.

Joshua Grossman, Zhiyuan Lin, Hao Sheng, Johnny Tian-Zheng Wei, Joseph J Williams, and Sharad Goel. 2019. Mathbot: Transforming online resources for learning math into conversational interactions. *AAAI 2019 Story-Enabled Intelligence*.

Yuxi Li. 2022. Reinforcement learning in practice: Opportunities and challenges.

Arcanjo Miguel Mota Lopes, José Francisco de Magalhães Netto, Ricardo AL de Souza, Andreza Bastos Mourão, Thais Oliveira Almeida, and Dhanielly PR de Lima. 2019. Improving students skills to solve elementary equations in k-12 programs using an intelligent tutoring system. In *2019 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE.

Roger Nkambou, Jacqueline Bourdeau, and Valéry Psyché. 2010. Building intelligent tutoring systems: An overview. *Advances in Intelligent Tutoring Systems*, pages 361–375.

Hyacinth S Nwana. 1990. Intelligent tutoring systems: an overview. *Artificial Intelligence Review*, 4(4):251–277.

Thanh Long Phan. 2020. Chatgpt: Reinforcement learning for chatbots.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer.

Holger Schwenk and Jean-Luc Gauvain. 2005. Training neural network language models on very large corpora. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pages 201–208.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. 2021. Ethical and social risks of harm from language models (2021). *arXiv preprint arXiv:2112.04359*.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*.

Jianyi Zhang, Xu Ji, Zhangchi Zhao, Xiali Hei, and Kim-Kwang Raymond Choo. 2023. Ethical considerations and policy implications for large language models: Guiding responsible development and deployment. *arXiv preprint arXiv:2308.02678*.