Automatically Generating Personalized Educational Exercises

Sabina Elkins

School of Computer Science
McGill University
Montreal, Canada

August 2023

A dissertation submitted to McGill University in partial fulfillment of the requirements for the degree of Masters of Science.

Abstract

With every passing year, artificial intelligence is increasingly integrated into education. The natural language processing task of question generation is a prime candidate for practical use in education, as questions are a main pedagogical intervention seen in both online and in-person learning. Pedagogical research emphasizes the power of personalized learning to greatly improve student understanding and performance. Finding improved ways to apply and adapt personalized question generation to education creates a useful tool to improve an instructor's content creation experience and a student's learning experience.

To demonstrate the power of personalized automatically generated questions, this thesis shows that personalized question variants help students to learn more effectively *and* that it is possible to generate such questions with high enough quality to be judged as useful by teachers. First, an experiment where students at different levels of subject proficiency are provided variants of a given question suitable for their needs demonstrates improved student learning gains, using questions written by a domain expert and an experimental A/B test. The results demonstrate that level-targeted linguistic realizations of questions positively affect learning outcomes for students. Then, educational question generation is explored using controllable text generation by large language models. A human evaluation is conducted with real teachers to assess the quality and usefulness of generating questions in question taxonomies, whose different levels can reflect the needs of different students. The questions generated are high quality and sufficiently useful, showing their promise for widespread use in the classroom setting. All in all, the value of personalized questions to student's learning is demonstrated, and then a robust approach to generating such questions is proposed and assessed.

Abrégé

Au fil des années, l'intelligence artificielle s'intègre de plus en plus dans l'éducation. La génération des questions est un candidat idéal pour une utilisation pratique dans la domaine d'éducation, car les questions sont une intervention pédagogique majeure observée dans l'apprentissage en ligne et en personne. La recherche pédagogique souligne le pouvoir des questions personnalisées pour améliorer la compréhension et les performances des étudiants. Trouver des moyens améliorés d'appliquer et d'adapter la génération de questions personnalisées peuvent créer un outil utile pour améliorer l'expérience de création de contenu d'un enseignant et l'expérience d'apprentissage d'un étudiant.

Cette thèse montre que les variantes de questions personnalisées aident les étudiants à apprendre de manière plus efficace *et* qu'il est possible de générer ces questions avec une qualité suffisamment élevée pour être jugées utiles par les enseignants. Tout d'abord, on fait une expérimente dans laquelle les étudiants de différents niveaux de compétence reçoivent des variantes d'une question adaptés à leurs besoins. Ils démontrent une amélioration des gains d'apprentissage des étudiants, avec des questions rédigées par un expert du domaine et un test A/B expérimental. Les résultats démontrent que les réalisations linguistiques des questions qui sont faites pour les niveaux différents ont un impact positif sur les résultats d'apprentissage des étudiants. Ensuite, la génération de questions éducatives est explorée en utilisant la génération de texte contrôlée par de grands modèles de langage. Une évaluation humaine est réalisée avec de vrais enseignants pour évaluer la qualité et l'utilité de la génération de questions dans des taxonomies de questions, dont les différents niveaux peuvent refléter les besoins des différents étudiants. Les questions générées sont de haute qualité et suffisamment utiles, ce qui montre leur promesse pour l'utilisation en classe. En tout, la valeur des questions personnalisées pour l'apprentissage des étudiants est démontrée, puis une approche robuste pour générer de telles questions est proposée et évaluée.

Previously Published Material

This thesis contains previously published work from the author:

- Elkins, S., Kochmar, E., Belfer, R., Serban, I., & Cheung, J. C. (2022, July). Question Personalization in an Intelligent Tutoring System. In Artificial Intelligence in Education. Posters and Late Breaking Results: 23rd International Conference, AIED 2022, Durham, UK, July 27–31, 2022, Proceedings, Part II (pp. 586-590). Cham: Springer International Publishing.

- Elkins, S., Kochmar, E., Serban, I., & Cheung, J. C. (2023, June). How Useful Are Educational Questions Generated by Large Language Models?. In International Conference on Artificial Intelligence in Education (pp. 536-542). Cham: Springer Nature Switzerland.

This thesis' author led the work in both of the above papers. The work in "Question Personalization in an Intelligent Tutoring System" can be seen in Chapter 3, with respect to the assessment of the effects of the linguistic realization of questions on student success while learning with an intelligent tutoring system. The work in "How Useful are Educational Questions Generated by Large Language Models?" can be seen in Chapter 4, specifically Section 4.3, where real teachers perform annotations to validate the usefulness of pedagogical questions generated automatically.

Acknowledgments

I would like to take this opportunity to thank a collection of individuals and organizations who have enabled my successful completion of this thesis.

First and foremost, a huge thank you to my wonderful supervisor, Prof. Jackie Chi Kit Cheung. His support and advice has been invaluable throughout my degree, and the work demonstrated in my thesis. I am certain that I would not have completed this research without his help and encouragement. Alongside Prof. Cheung, I would like to express my sincere appreciation to my peers in the Reasoning and Learning Lab at McGill University and at Mila, especially Cesare. Their camaraderie has been an amazing source of support.

I would also like to thank my industry supervisors, Ekaterina Kochmar and Iulian Vlad Serban. Their expertise and insightful feedback have shaped the direction of my research and enriched my academic experience. I am very grateful for their guidance and mentorship. I would also like to extend my gratitude to the rest of my colleagues at Korbit where I have had the privilege of working through an industry collaboration for the course of my degree.

I would also like to acknowledge the support of Mitacs, who, alongside Korbit, graciously provided the financial support necessary for the successful completion of my degree. Additional organizations who I'd like to thank for their support include the Canada CIFAR AI Chair program, and Mila.

Finally, I would like to express my deepest gratitude to my family and friends, especially Dion, Joely, Nicola, and Jonathan. I am grateful for their encouragement, patience, and belief in my abilities. Their presence has provided me with the strength and motivation to overcome challenges and complete this thesis.

To all those who have contributed to this thesis in various ways, whether large or small, I extend my heartfelt thanks. Your contributions have made a profound impact on this work and my personal growth. Thank you.

Contents

1 Introduction **1**

1.1 Thesis Statement 3

1.2 Objectives 3

1.3 Structure 4

2 Literature Review **5**

2.1 Pedagogy 6

2.1.1 Merits of 1-on-1 Tutoring 7

2.1.2 Importance of Questions 9

2.1.3 Question Taxonomies 10

2.2 Intelligent Tutoring Systems 14

2.2.1 The History of ITSs 14

2.2.2 The Potential of ITS 15

2.2.3 Case Study ITS: `Korbit` 16

2.3 Large Language Models 18

2.3.1 Theoretical Background of LLMs 20

2.3.2 Transformer-Based LLMs 23

2.3.3 Prompting LLMs 24

2.3.4 Text-to-Text Transfer Transformer Model (`T5`) 25

2.3.5 Generative Pre-trained Transformer (GPT) 25

2.3.6 Controllable Text Generation 27
2.4 Question Generation 28
2.4.1 A Brief History of Educational Question Generation 30
2.4.2 Educational Question Generation with LLMs 32

3 Personalizing the Learning Experience **36**
3.1 Handwritten Question Variants 37
3.2 Choosing Question Variants for Students 40
3.3 A/B Test 42
3.4 Conclusion 45

4 Generating Educational Questions **46**
4.1 Automatically Generating Question Variants 47
4.1.1 Sentence Simplification 48
4.1.2 Elaboration Generation 51
4.1.3 Paraphrasing 53
4.2 Preliminary Experimentation for QG with LLMs 55
4.2.1 Generation and Assessment Procedure 56
4.2.2 Parameter: Context Length 59
4.2.3 Parameter: Context Domain 60
4.2.4 Parameter: Shot Setting 61
4.2.5 Parameter: Control Elements 62
4.3 Teacher's Opinions on the Usefulness of QG with LLMs 63
4.3.1 Generation with `InstructGPT` 64
4.3.2 Methodology for Usefulness Study 64
4.3.3 Results of Usefulness Study 68
4.4 Conclusion 72

5 Conclusion **73**

5.1 Limitations 74

5.2 Future Directions 75

A Variant Generation Examples **76**

B Generating Educational Questions with `InstructGPT` **79**

B.1 Question Types and Taxonomies 79

B.2 Contexts 80

B.3 Few-Shot Examples 81

B.4 Examples of Generations 82

Bibliography **85**

List of Figures

1.1 Academia Group's survey had 270 subjects, of which 104 (39%) reported using `ChatGPT` for course-related work. This graph is a breakdown of their self reported use cases (Milian and Janzen, 2023). 2

2.1 Various examples of `Korbi`'s web interface, and teaching interventions. 17
2.2 A simple form of an artificial neural network. 18
2.3 Visualizations of the basic encoder-decoder model, and of attention. . . 20
2.4 A basic visualization of a Transformer architecture. Note that this is an oversimplification. 22

3.1 An exemplary question adapted to different difficulty levels while retaining the same correct answer/group of correct answers (the different pluralities is small enough to be overlooked). 38

4.1 Generation Prompt Template (*one-shot* template) 57
4.2 Visualizations of the *context length* and *context domain*. Significant difference using Student's t-test and $\alpha = 0.05$ is marked by an asterisk. 60
4.3 Visualizations of the *shot setting* key results. Significant differences using Student's t-test and $\alpha = 0.05$ are marked by an asterisk. 62
4.4 Demographics and educational experience of the participants. 65
4.5 Visualizations of the *usefulness* and *adherence* metrics. 71

List of Tables

3.1 Mean variant scores from human experts, and average word counts by level. Arrows indicate better scores for strictly directional metrics. *Difficulty*, *Fluency*, and *Meaning Preservation* are on a scale from 0 to 5. 39
3.2 Features considered in next-exercise-success prediction model. N.b., a topic on the Korbit platform is a broad category of material, such as 'Probability' or 'Deep Learning'. 41
3.3 Test results. Arrows indicate better scores for strictly directional metrics. Metrics marked with * have a statistically significant difference between them at the $\alpha = 0.05$ level by a Student's t-test. 44

4.1 `ACCESS` simplified question variant annotated results. Arrows indicate better scores for strictly directional metrics. 50
4.2 Elaborated question variant annotation results. Arrows indicate better scores for strictly directional metrics. 53
4.3 `Parrot` paraphrased question variant annotation results. Arrows indicate better scores for strictly directional metrics. 54
4.4 Summary results from the qualitative assessment of `InstructGPT` controllable generated educational questions. Arrows indicate better scores for strictly directional metrics. 58

4.5 The quality metrics' mean (μ), standard deviation (σ), and observed agreement (i.e., % of the time the annotators chose the same label). The *n* for the *usefulness* metrics are twice as large because all annotators rated them, unlike with the quality metrics. Arrows indicate better scores for strictly directional metrics. 70

A.1 Examples of exercise variants simplified with ACCESS. 76
A.2 Examples of elaboration generation exercise variants. 77
A.3 Examples of paraphrased exercises using Parrot. 77

B.1 Question types used in the preliminary experiment in Section 4.2 79
B.2 Examples of contexts used for educational question generation. 81
B.3 Examples of hand-crated questions used for few-shot learning within the educational question generation prompts. 82
B.4 Examples of automatically generated educational questions using few-shot learning and InstructGPT. 82

List of Acronyms

AI	artificial intelligence
AIED	artificial intelligence in education
NLP	natural language processing
ML	machine learning
QG	question generation
ITS	intelligent tutoring systems
MOOC	massive open online course
ZPD	zone of proximal development
LLM	(transformer-based) large language models
GPT	generative pretrained transformer
T5	text-to-text transfer transformer model
CTG	controllable text generation
CWI	complex word identification
TF-IDF	term frequency inverse document frequency

Chapter 1

Introduction

Changes in the face of advancements with artificial intelligence (AI) can be seen in virtually every field possible, from finance to the fine arts. Importantly, the use of AI-based technologies is becoming pervasive in education (Kasneci et al., 2023; Chen et al., 2020). This can be both a detriment and an advantage. For example, consider the uproar caused by the release of `ChatGPT` in November 2022. Initial reactions by students, teachers, and educational policy makers varied, from media coverage with intentionally provocative titles like "The College Essay Is Dead" and worries about the technology enabling students to cheat (i.e., by plagiarising text the model outputs) (Marche, 2022; Milian and Janzen, 2023) to excitement about the novel applications that `ChatGPT` enabled Kasneci et al. (2023). A recent anonymous survey by the Academia Group on Canadian post secondary student's use of `ChatGPT` produced the breakdown seen in Figure 1.1. From this survey at least, it appears that only a small minority of students are using `ChatGPT` to complete whole or partial assignments. It is important to mention with a study such as this that there exists a possibility of bias resulting from students not wanting to admit to such use of `ChatGPT`. Regardless, 61% of students do not report using `ChatGPT` at all.

In a case study concerning the use of `ChatGPT` in education, Kasneci et al. (2023)

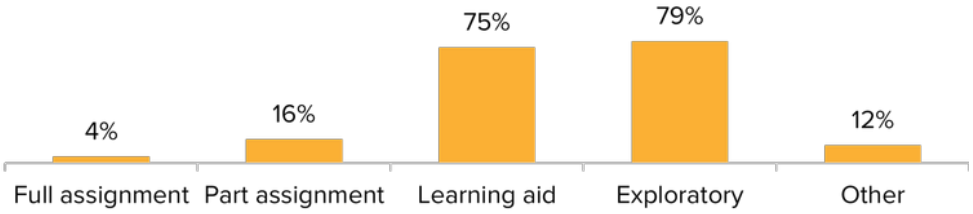


Figure 1.1 Academia Group's survey had 270 subjects, of which 104 (39%) reported using `ChatGPT` for course-related work. This graph is a breakdown of their self reported use cases (Milian and Janzen, 2023).

outline enticing benefits the technology can provide, including the creation of educational content, increasing student engagement, personalizing the learning experiences of students, and more. These benefits are echoed in a more general study by Chen et al. (2020), where the authors conduct a literature review of academic works concerning artificial intelligence in education (AIED). They conclude that AI has been able to personalize educational content and curriculums to improve student engagement and learning experiences, and that instructors have been able to use AI to improve both the efficiency and quality of their educational content and their evaluations of students. Theses authors admit that potential of AI-based technologies comes at a cost. Kasneci et al. (2023) warn of `ChatGPT`'s potential for bias and hallucination, it's ability to help students cheat, the creation of strong reliance on the model, and other challenges. However, the authors pose suggestions to turn these challenges to advantage, by teaching students about the pros and cons of AI applications, and developing their critical thinking skills.

1.1 Thesis Statement

Regardless of any individual's beliefs about the merit of using AIED, its increasing popularity and use indicate that it is here to stay. Thus, researchers have the responsibility to assess the best ways to use such technologies for good and overcome the aforementioned challenges. This thesis focuses on how to use AI to generate personalized educational content, specifically educational questions. The personalization aspect is critical, because most current approaches to generating educational questions do not focus on it (Kurdi et al., 2020).

The hypothesis is that learning better ways to apply and adapt question generation for personalization in the educational domain can improve both the learning and the teaching experience. The experiments outlined in subsequent chapters show how beneficial personalized questions can be for learning outcomes and how natural language processing (NLP) techniques can be applied to generate diverse and high quality educational questions; thus showing how automatically generated educational questions can improve student learning experiences through personalization and reduce the load placed on teachers to generate educational content.

1.2 Objectives

The work in this thesis aims to provide evidence for two key aspects of the hypothesis. First, high-quality, level-adapted, exercise variants are shown to improve learning for students with an intelligent tutoring system (ITS), which is an AI-based technology for online tutoring (see Section 2.2 for more details). Demonstrating the increase in learning gains students experience with personalized exercises is a critical motivation to improve and validate work in the NLP task of question generation (QG), as it pertains to education. So naturally, the second aspect explored concerns the automatic generation of personalized questions. AI models, specifically large language models (see

Section 2.3 for more details), can generate different types of questions from a given context that teachers judge as sufficiently useful for the classroom. These different types of questions cater to the needs of unique students, and as such can be used to personalize their experience. Showing both of these demonstrates the value of personalized educational questions and a teacher-approved approach to generating them. In doing so, it is possible to encourage their use in various classroom settings, from automatic applications in an online ITS to a content generation tool ready to be put in teachers' belts.

1.3 Structure

In the following chapters, the different aspects of personalization of educational question generation are covered. In Chapter 2, the key research in related pedagogical theory, intelligent tutoring systems, large language models, and question generation is explored. In Chapter 3, an experiment to demonstrate the merit of personalization of questions is explained and it's outcomes are analyzed. In Chapter 4, an approach to controllable question generation is discussed, and a human assessment is conducted by real teachers to assess the educational potential of the generated candidates. Finally, in Chapter 5 reiterates the hypotheses and findings discussed in the whole thesis, and mentions future directions for this research.

Chapter 2

Literature Review

In recent years, the use of AIED has evolved in tandem with the development of NLP technologies that attempt to process and pull meaning from language data. NLP and machine learning (ML) have been used to develop conversational agents which aim to provide personalized tutoring and support to students, analyze large datasets of student performance data to identify patterns and predict outcomes, and generate various kinds of educational content automatically. While the successes and advances in this area continue to pile up, there is still much room for improvement. It is crucial for researchers to consider the existing corpus of research in both education and NLP that can direct the way forward at the intersection of these fields.

This chapter contains a literature review of the related work to this thesis's focus on personalization and automatic generation of educational content. It is by no means a complete literature review of the intersection of education and AI, or educational content generation, or even a complete list of the research that was conducted for this thesis. However, it covers all the theoretical background required to understand the work presented. First, Section 2.1 discusses some pedagogical theory related to the merits of personalization of education and the value of questions. Next, Section 2.2 defines ITS and explains their potential for personalized, scalable, online education.

Then, Section 2.3 covers the basics of large language models (LLMs), the details of T5 and the GPT family, and the applications of these models seen in this thesis. Finally, Section 2.4 explores the NLP task of question generation (QG), and related work in applying QG to the educational domain.

2.1 Pedagogy

Pedagogy is the study of education, encompassing everything from the abstract goals of education to the precises methods to achieve such goals (Peel, 2023). This field of study is often considered to encompass both educational philosophy and the act of teaching. In other words, pedagogy can be described as having two key goals. First, to understand how students learn. And second, to optimize instructional materials and activities to improve student learning.

Pedagogy is an expansive field, considering its diversity of topics and its age. By comparison, NLP and AI are young fields of study. This is especially true for the use of AIED; with computer-aided instruction having only been introduced in the mid-1900s, and AI's inclusion happening after that (Chen et al., 2020). As a consequence it can be difficult to isolate the key topics needed to inform AIED, and even harder to choose which theories to subscribe to. Accordingly, the following subsections touch on some aspects of pedagogical research that are relevant and informative for this thesis. They are not a complete list of the related educational theories to this work, or of theories that could inform how educational content should be responsibly generated. Yet, the included works have been considered and selected on the basis of their soundness of argumentation, relevance to the goals of this thesis, recency and popularity amongst pedagogical scholars. Firstly, Section 2.1.1 discusses the merits of 1-on-1 tutoring, or personalized learning, over the classroom model. This is a critical point to support the creation and adoption of ITSs (whose theory is outlined in Section 2.2). Secondly,

Section 2.1.2 explains how vital not only questions, but the *right kind* of questions, are to the process of learning. Finally, Section 2.1.3 introduces some different methodologies from pedagogical theory to classify questions into taxonomies.

2.1.1 Merits of 1-on-1 Tutoring

Based solely on intuition, there is already a strong case for one-on-one tutoring over a large classroom setting. Tutoring can provide students with pedagogical content tailored to their learning styles and enable them to have interactions with a tutor to uncover and patch misconceptions in their understanding. This can also be described as a tutor's ability to personalize a student's learning experience. In this context, personalization means that pedagogical exercises, questions, feedback, explanations and more are tailored to the unique needs and learning style of a given student. It is easy to see how an experienced and skilled tutor might achieve this while teaching one student at a time. Conversely, in a classroom setting teachers must attempt to cater to the needs of a large group of students with varying abilities and learning styles; resulting in an often imperfect match between the teaching style and each student.

This intuitive explanation is confirmed by numerous studies stretching over decades of educational research. The research shows the merits of one-on-one tutoring over traditional classroom-style learning with respect to both academic performance and student's perceptions of the learning experience:

1. Bausell et al. (1972) performed a detailed comparison experiment with cohorts of fourth grade students who either learn material with a tutor or in a classroom with their peers. The authors reported statistically significant improvements in tutored student's scores on a post quiz. The authors also take into account student's levels (using their grades up to that point) and the trends they see hold across all student levels.

2. Bloom (1984) compared settings of one-on-one tutoring and one-to-30-student classrooms. The authors concluded that an average tutored student has a test result which is a whole two standard deviations above that of a classroom-taught student. In this work the authors go on to explore group learning methods that improve the learning outcomes towards those of one-on-one tutoring, with the goal of balancing resource constraints with improved learning.

3. Hattie (2012) discusses optimal teaching practices, including referencing studies demonstrating that students receiving one-on-one tutoring outperform their peers. These experiments considered tutoring as an addition to classroom teaching, as opposed to a replacement. At this point, computer assisted teaching methods were more commonplace.[1] The author argues that there is substantial evidence that the personalization achieved in one-on-one tutoring which improves student learning outcomes, can also be achieved with technology.

4. St-Hilaire et al. (2022) demonstrated the merits of personalized education in an entirely online context. Their experiment compared students' learning gains when learning through a massive online open course (MOOC) and when learning with an ITS. A MOOC is essentially a classroom-style online course where all students receive the same video lectures and exercises, despite not being in the same physical room. Conversely, the ITS used in this study personalizes the content a student receives using AI (this particular ITS's approach is explained in detail in Section 2.2.3). The results show statistically significant improvements of those students learning one-on-one with an ITS over those learning with a classroom-style MOOC.

These four points cover a breadth of arguments from older foundational works in the study of education up to newer arguments as pedagogical research shifts to encompass

[1] See Section 2.2 for a more comprehensive historical background of computer-aided instruction and ITSs.

online education. They all demonstrate that one-on-one tutoring improves learning gains and experiences for students.

Unfortunately, one-on-one tutoring is very resource expensive. Having to pay a tutor is not something everyone can afford, and is certainly not sustainable for government-funded education systems like that which we have in Canada. This has led researchers to look at ways to reproduce these effects; from replicating one-on-one tutoring benefits in the classroom setting (Bloom, 1984) to, more recently, building AI systems to automatically create personalized learning experiences (St-Hilaire et al., 2022). Section 2.2 explains in more detail the use of AI to create ITSs that are automated tutors who can subvert these resource constraints.

2.1.2 Importance of Questions

Questions play a crucial role in education. In an active learning environment, they have the power to stimulate critical thinking, encourage active engagement, and help students to retain information more effectively in an effort to promote a deeper understanding of the material being taught. Questions are also invaluable as a post-lecture exercise to challenge students to analyze and synthesize information, make connections between concepts, and draw conclusions based on evidence. Additionally, questions can provide valuable feedback to teachers about how well students are understanding the material. By asking questions, teachers can identify areas where students may be struggling and adjust their methods to address these challenges. For these reasons and more, questions are imperative for learning and are a fundamental pedagogical intervention used by teachers and tutors.

While it is clear that questions are useful, it is not so clear how teachers should optimize their questions to promote learning. There are many existing, and often conflicting, viewpoints in this vein. For instance, even within the relatively limited works cited in this thesis, each of Taylor (1962); Ashton-Jones (1988); Graesser and Person

(1994); Hattie (2012) have a stated opinion on how to ask the best questions, which agree with each other only somewhat. In order to sort through the conflicting opinions, it can be beneficial to try and isolate the teacher's goal when asking questions. Often, the goal is not necessarily to have students answer correctly, but instead to challenge them in order to encourage a robust understanding of the presented material. The level of challenge necessary can be informed by a foundational theory to educational psychology called the zone of proximal development (ZPD). ZPD was introduced by the Russian psychologist Lev Vygotsky in 1931 (Hedegaard, 2012). This concept is meant to describe the difference between what a student is able to do independently and what they can achieve with guidance and support from a teacher (Hedegaard, 2012). ZPD is not a fixed range, but rather a dynamic concept that changes over time as learners gain new skills and knowledge. Vygotsky believed that learning occurs when learners are challenged to reach beyond their current level of understanding and receive support and guidance from a teacher. In other words, a student needs to be within their ZPD in order for learning to occur. By providing just the right amount of challenge and support to individual learners, a teacher can help learners bridge the gap between their current abilities and their potential abilities.

2.1.3 Question Taxonomies

For argument's sake, let us assume that the goal of question asking is to achieve student's individual ZPDs. Even in this case, it is hard to optimize what attributes of a question make it optimal for learning. Unfortunately, there is no 'silver bullet' question type that is capable of increasing student learning outcomes and satisfaction by itself (Hrastinski et al., 2021). Instead, questions must be mapped to the situation along a variety of axes. A question should be linked to a specific teaching goal, such as memorization of important facts or critical thinking about a presented argument. It can also be improved by reflecting a student's level, such as their vocabulary, background

knowledge, or other strengths and weaknesses they have. As such, different types of questions are required for different situations. There is a rich history in pedagogical theory of research into taxonomies to organize questions into groups. These question taxonomies can help teachers and students alike to analyze what questions are appropriate in any given situation, and their different purposes. The following paragraphs will introduce three of these question taxonomies used later in this work.

Bloom's Taxonomy The most famous question taxonomy is Bloom's taxonomy, due to the impact Bloom himself had on American educational practice and research (Lasley, 2023). Actually, Bloom's taxonomy is a framework for categorizing *learning objectives* and identifying different levels of cognitive complexity in educational objectives (Krathwohl, 2002). There are alternative approaches to classify questions by their learning objectives, such as in work by Day and Park (2005), but Bloom's work is the most popular to date. Bloom's taxonomy is often applied to questions which themselves have learning objectives based on what they ask a student to do: recall information, think critically, be creative, etc.. Bloom's taxonomy was first created by Benjamin Bloom in the 1950s and has since been revised and expanded upon (Bloom, 1956; Krathwohl, 2002). The newest version of the taxonomy contains six levels of learning. These are arranged in a hierarchical order from 'lower-order thinking skills' to 'higher-order thinking skills':

1. **Remembering**: Retrieving from memory previously learned information or facts (e.g., a term, a concept, a definition, a formula).

2. **Understanding**: Demonstrating comprehension of the meaning of the information (e.g., explaining ideas in one's own words, identifying cause-and-effect relationships, comparing two similar ideas).

3. **Applying**: Using learned information in a novel or different situation (e.g., solv-

ing a problem or applying a formula to a real-life scenario).

4. **Analyzing**: Breaking down material into its component parts, identifying patterns or connections between different ideas, and drawing conclusions.

5. **Evaluating**: Giving opinions, making judgments, or interpreting the value or quality of information or arguments (e.g., evaluating the strengths and weaknesses of an argument).

6. **Creating**: Generating original or innovative ideas by combining parts of the material in a different way than presented (e.g., designing a new solution to a problem or developing a new theory).

In practice, this categorization can assist teachers in designing instructional activities and materials that target specific learning goals. They can also be used to guide controllable generation of pedagogical content, as will be seen in the later chapters of this thesis.

Difficulty-Level Taxonomies In place of learning goals, questions can be classified by difficulty level, producing an aptly named difficulty-level taxonomy. This kind of taxonomy is seen in various different pedagogical works. For example, specifically in AIED, an answer-type taxonomy is seen in work by Pérez et al. (2012). The authors use a three-tier difficulty-level taxonomy in their attempts to create an expert system to automatically classify the difficulty of questions. Such a taxonomy can also be used in a looser sense, where questions are mapped to a continuous value according to their difficulty (Tan and Othman, 2013).

These examples all have a different number of difficulty levels considered, and different criteria outlining what makes a question easy or hard. This follows from the fact that students will have different perceptions of the difficulty of any content, and

struggle or excel at different topics, or even concepts within a topic. This makes classification into a difficulty-level taxonomy, well, difficult. A simple strategy is to use three strata of difficulty, roughly mapping to **easy** (or **beginner**), **medium** (or **intermediate**), and **hard** (or **advanced**).[2] This separation is usually done with respect to student's scores on assessments (Tan and Othman, 2013) or on student (and/or teacher) perceptions of the questions (Pérez et al., 2012). Later chapters in this thesis will use these three difficulty categories to both assign and generate content fitted to the needs of individual students.

Answer-Type Taxonomies Another axis upon which to classify question taxonomies is their expected answer type, or the question's form. For example, (Day and Park, 2005) differentiate between five forms of questions. This classification process is much easier than the previous, but has a longer list of potential categories. Works using answer-type question taxonomies classify questions into groups such as:

- **Multiple Choice**: Questions where multiple answers are provided to the student, whose task is to isolate the correct solution(s).
- **Short-Answer**: Questions where the student is expected to provide a short textual response (such as a keyword or phrase).
- **Number**: Questions where the student is expected to provide a numerical answer, either through calculation or from memory.
- **True or False**: Questions where the student is expected to judge whether a provided statement is true or false.

The above examples only scratch the surface of potential classifications in an answer-type taxonomy. For instance, Graesser and Person (1994) include 18 different groups in

[2] As seen in a variety of use cases, such as Pérez et al. (2012) involving questions from an undergraduate engineering course, White and Iivonen (2002) involving web search questions, Vamsi et al. (2020) involving programming questions, and more.

their answer-type question taxonomy, but do not include any of those mentioned in the list above. It is obvious to say that different answer-type questions create different opportunities for learning. They are different tools teachers can use. But, it can be difficult to classify questions into such taxonomic levels due to the fact that the classifications cannot be easily reduced to an agreed-upon set of categories.

2.2 Intelligent Tutoring Systems

Intelligent tutoring systems (ITSs) are computer-based systems that provide personalized instruction and feedback to students, mimicking the role of a human tutor. ITSs have been implemented across a wide range of educational settings, spanning from K-12 classrooms (e.g., the math tutor by King et al. (2021) for 6th grade students) to professional training programs (e.g., SHERLOCK by Lajoie and Lesgold (1989), which trains Air Force pilots on electrical problems in F-15 jets). ITSs have also been developed for teaching a huge variety of topics, from programming languages (e.g., JavaTutor by Wiggins et al. (2015)), to spoken languages (e.g., a French language tutor by Khella and Abu-Naser (2018)). These examples represent only a small fraction of the numerous and diverse ITS projects that have been developed in recent years.[3] The following sections will outline a brief history of ITSs, the educational potential of ITSs, and a case study of the ITS that was used in the course of this thesis.

2.2.1 The History of ITSs

The idea of developing a computer-based tutoring system to augment learning has been around for decades. As computers started to become more accessible in the 1970s, the idea of computer-aided instruction (CAI) was popularized (Alkhatlan and Kalita, 2018). CAI is simply instructional material presented by way of a computer. At

[3]See Alkhatlan and Kalita (2018) for a more examples of ITSs from 2000 to 2018.

this time, major universities began using computers in educational settings, and these institutions, along with technology companies, began to support CAI programs and projects (Chambers and Sprecher, 1983). Through the 1980s, CAI morphed into something called *intelligent* computer assisted instruction (Larkin and Chabay, 1992). This shift marked the introduction of problem-solving capabilities into these rudimentary computer-teaching systems. These abilities improved in the 1990s, as ML and AI methods made their debut in educational computer systems (Larkin and Chabay, 1992). In the late 1990s and early 2000s, dialogue-based ITSs were introduced with systems like AutoTutor (Graesser et al., 2005). These ITSs act like chat-bots, conversing with a student to give them lessons, exercises, and feedback. While this is not the only type of ITS, it is the most common, due to its mimicry of a human tutor (Graesser, 2011; Alkhatlan and Kalita, 2018). Since then, ITSs have been steadily improving alongside the advent of new AI techniques.

2.2.2 The Potential of ITS

ITSs may have the potential to revolutionize education. Critically, they have the power to improve access to education, as many ITSs operate online and can be accessed by anyone with connection to the internet. However, their potential is not only in accessibility and scalability. Many ITSs have the goal of improving students' learning outcomes by adapting the instructional content and strategies to each student's individual needs and abilities (Alkhatlan and Kalita, 2018). An ideal ITS can provide personalized, adaptive, and data-driven instruction and feedback to individual learners. In other words they can use the information gained through a student's interactions with the tutor to give appropriate feedback at all points of problem-solving, and choose what material to teach next based on the student's knowledge gaps (Kochmar et al., 2020).

The benefits of ITSs have been examined by various researchers across different

contexts. Graesser (2011) conducted experiments using AutoTutor, an ITS designed to teach computer literacy and conceptual physics to students. The results of this study revealed an improvement of 0.8 standard deviations in learning gains among students using AutoTutor compared to those who simply read the same material. While this individual example is informative, it fails to speak to the success of ITSs generally. In that vein, VanLehn (2011) analyzed the outcomes of 87 previous studies and found that ITSs are almost as effective as human tutoring, with an effect size of 0.76 compared to human tutoring's 0.79.[4] These findings are corroborated in Kulik and Fletcher (2016), which reported a median improvement of 0.66 standard deviations in post-test scores across 50 ITS studies compared to control groups. Overall, these studies demonstrate the potential of ITS to significantly increase learning gains for students.

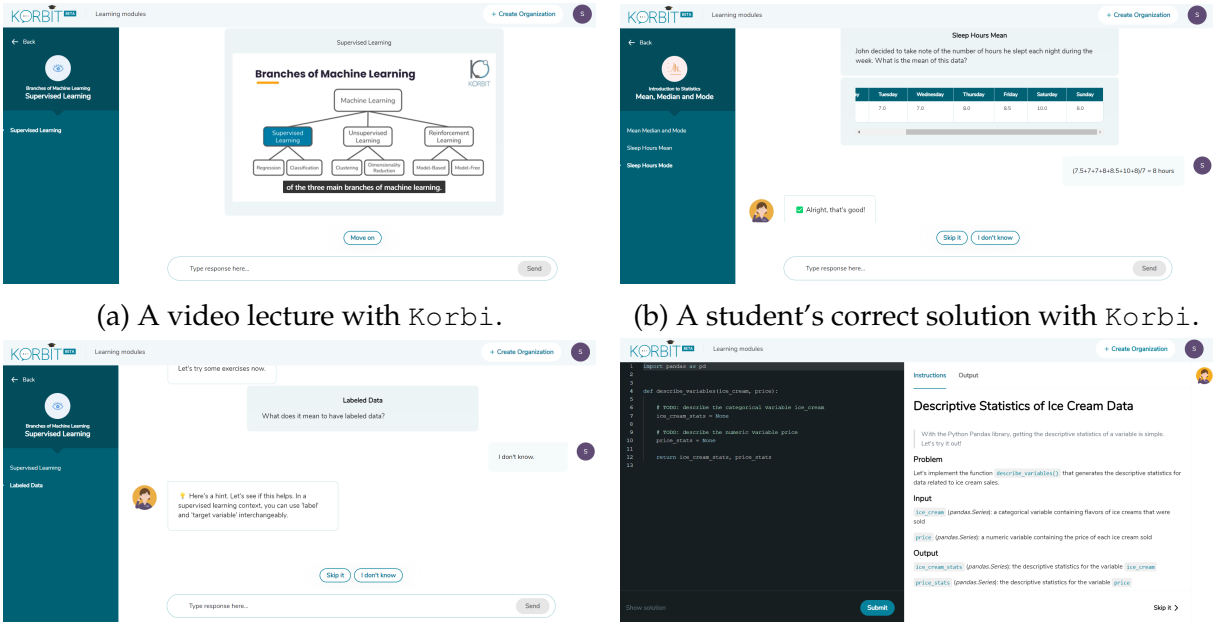
2.2.3 Case Study ITS: `Korbit`

Korbit Technologies is a Montreal-based start up company founded with support from AI researchers at the Montreal Institute for Learning Algorithms (MILA) and Cambridge University. Their mission is to "provide personalized, high-quality training for millions around the world at a low cost, anytime and anywhere" (Korbit), by way of innovation with ITS technology. Korbit's first AI tutor, `Korbi`, is a dialogue-based ITS, which teaches students online in a chat-based setting.[5] As a student interacts with `Korbi`, the teaching materials they receive are selected using a collection ML and NLP techniques (Serban et al., 2020; Kochmar et al., 2020; St-Hilaire et al., 2022). These 'teaching materials' include a variety of possibilities: video lectures, project-based learning modules, socratic tutoring, interactive problem solving exercises, coding exercises, personalized feedback, and more. Figure 2.1 shows a few examples of the interface. `Korbi` has taught over 20,000 users about mathematics, statistics, data

[4] N.b., The effect size is compared to a control group that received no additional tutoring

[5] N.b., `Korbi` is Korbit's V1 product. The company is developing a new ITS solution, in other words their product V2, but the work related to this thesis was in tandem with their first product.

science, machine learning, and other related topics (St-Hilaire et al., 2022).



(a) A video lecture with `Korbi`.

(b) A student's correct solution with `Korbi`.

(c) `Korbi` providing a hint to a lost student.

(d) A programming exercise with `Korbi`.

Figure 2.1 Various examples of `Korbi`'s web interface, and teaching interventions.

A user study conducted in 2020 showed that for over 600 users, learning with `Korbi` results in an average 39.14% increase in overall learning gains as compared to a MOOC (Serban et al., 2020).[6] This increase is based on users who performed quizzes through their learning process to assess their understanding; the improvement is the average score increase from the beginning to the end of the experiment. Since 2020, these results have been improved, with research by the company showing 49.24% higher learning gains in 2021 (St-Hilaire et al., 2021) and an astounding 90% in 2022 (St-Hilaire et al., 2022).[7]. These impressive results indicate that `Korbi` is a high performing ITS, taking steps towards the potentials outlined in Section 2.2.2. Chapter 3 will explain an experiment conducted with students using `Korbi`, to further the

[6]Introduced in Section 2.1.1, a MOOC is a massive open online course that generally consists of video lectures and associated static exercises.

[7]N.b., both of these studies are conducted similarly to the original: in comparison to MOOCs.

frontiers of personalization within ITSs.

2.3 Large Language Models

Artificial neural networks are ML algorithms inspired by the structure and function of a brain. They work by transforming data across a series of interconnected nodes using learned weights, and then applying an activation function to produce an output (Goodfellow et al., 2016). A basic neural network consists of a input layer, a hidden layer, and an output layer, shown in Figure 2.2. The circles represent layers in the network, and the arrows represent the weights applied to the data as it travels through the network. The input layer receives the initial numerical data, and passes weighted

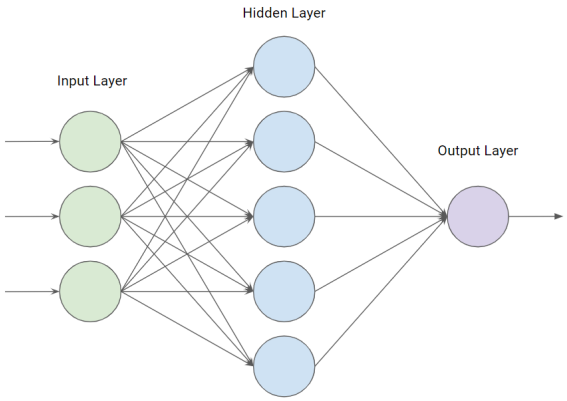


Figure 2.2 A simple form of an artificial neural network.

values along the arrows to the various nodes in the hidden layer. The hidden layer does the same to the output layer, where an activation function, g, is applied. This activation function can be a variety of different functions, common ones include the linear function, logistic function, and hyperbolic tan. The entire network learns a function f, which maps from the input data X to the output y:

$$y = f(X;\theta) = W_2 g(W_1^T x + b_1) + b_2$$

where θ represents all of the learned parameters: W_1 is the weights of the input layer, W_2 is the weights of the hidden layer and b_1 and b_2 are bias values added to the hidden layer and output layer, respectively. Note that the input and output data are always numerical for a neural network. In the case of textual data, these numerical representations are vectors representing the words, or tokens, in the data.

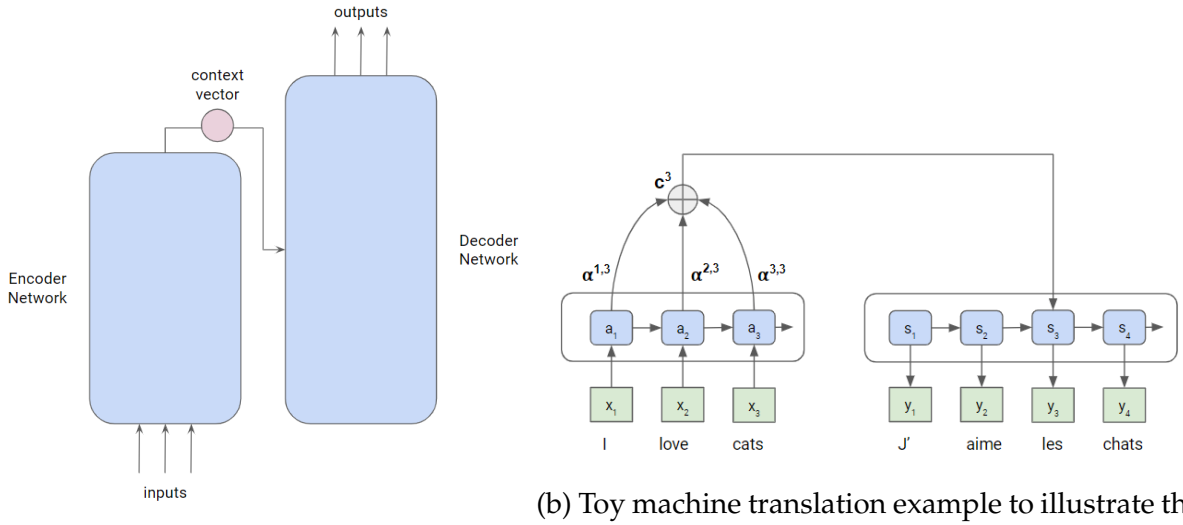
More complex versions of artificial neural networks where there is more than one hidden layer, are referred to as deep neural networks. These are the focus of much of the research in NLP today. The term 'large language models' (LLMs) generally refers to massive deep neural networks, typically with billions of parameters. These models are trained through exposure to trillions of examples of language use, enabling the model to internalize patterns, relationships, and rules within the language. This exposure can be a variety of different training objectives. A simple and common example is next token prediction, where the model learns to output the most likely next word given the prior context. Many LLMs are multi-purpose, meaning they are able to excel at a wide range of NLP tasks such as translation, text summarization, sentiment analysis, question generation, and more. Often they can do this because they treat all tasks as text-to-text; where the input is a piece of text structured so that the most likely completion outputted by the LLM is the desired output. As a result of their extensive abilities, these models have been applied in various real-life scenarios such as search engines, chatbots, content creation tools, and more.

In recent months, the popularity of LLMs has exploded. Outside of the field of NLP, this can be attributed to the release of `ChatGPT` by OpenAI in November 2022 (see Section 2.3.5 for more details on this LLM). The quality of outputs by `ChatGPT` was so striking that the model became famous beyond the NLP community and sparked global interest. These developing technologies are hugely exciting, with some researchers even going so far as to use the diverse abilities of theses systems as evidence to claim we are approaching artificial general intelligence (Bubeck et al., 2023). How-

ever, even the most advanced of these models are not without drawbacks. LLMs have internal faults such as bias introduced through their training data, the hallucination of non-factual content, and difficulties with basic arithmetic (Bubeck et al., 2023). There are also valid concerns about how they might be used for ill: from spreading compelling fake news with audiovisual deepfakes (Horvitz, 2022), to enabling plagiarism and cheating by students (Kasneci et al., 2023). Discussion of these excitements and concerns are vital to understanding the current landscape of NLP and how LLMs are affecting it. Unfortunately, they cannot be addressed within the scope of this work. Instead, sections 2.3.1 to 2.3.6 (as well as 2.4) will simply discuss the technology in more detail as it pertains to the work in this thesis.

2.3.1 Theoretical Background of LLMs

The rise of NLP's current deep learning paradigm can be traced back quite far in the history of research into neural networks. Without going too far back into the history, a



(a) An encoder-decoder architecture.

(b) Toy machine translation example to illustrate the attention mechanism.

Figure 2.3 Visualizations of the basic encoder-decoder model, and of attention.

good place to start for an intuitive explanation is with the idea of attention introduced

by Bahdanau et al. (2014). First introduced for use in machine translation, attention is an approach to help a deep neural network focus on certain parts of an input sequence when generating an output sequence. One use of attention is in encoder-decoder networks. The visualization in Figure 2.3a, shows the two base neural networks used in this architecture: the encoder network, whose hidden states we represent by a, transforms an input sentence X into a context vector c; and the decoder network, whose hidden states we represent by s, is fed the context vector and transforms it into an output sequence Y.

An example of the attention mechanism is depicted in Figure 2.3b. Attention allows the decoder network to selectively focus on different parts of the input sequence ($i \in 1, \ldots, |X|$) at each decoding step ($j \in 1, \ldots, |Y|$) as opposed to having equal importance placed on the entire input. The context vector is calculated using weights, $\alpha^{\langle i,j \rangle}$, which represent how much attention the network places on position i at decoding step j. These weights are learned by a feed forward neural network[8], f, which at decoding step j is given the decoder's previous state (s_{j-1}) and the encoder's state at position i (a_i):

$$\alpha^{\langle i,j \rangle} = \mathit{softmax}(f(s_{j-1}, a_i))$$

Then, the weights $\alpha^{\langle 1,j \rangle}, \ldots, \alpha^{\langle |X|,j \rangle}$ are used to compute a weighted sum of the encoder's hidden states, i.e., the context vector c_j:

$$c_j = \sum_i \alpha^{\langle i,j \rangle} a_i$$

The context vector is then given to the decoder network, let's call it g, to generate s_j:

$$s_j = g(s_{j-1}, c_j)$$

[8]N.b., A feed forward neural network is simply a neural network where the connections between the nodes do not form a cycle.

From here, the discrete output y_j is computed using a softmax function. This is repeated until the output reaches a stopping condition.

Vaswani et al. (2017) expanded on the original idea of attention by creating two variations: self-attention and masked-attention. Self-attention is where the model attends to words within the same sentence. Masked-attention is where the next tokens in the sentence are masked (i.e., hidden). Additionally, Vaswani et al. (2017) use multi-head attention, which is just a collection of self-attention mechanisms. With these attention mechanism updates, the authors introduce a now famous deep neural network architecture: the Transformer. A Transformer's encoder applies multi-head attention to the input sequence, creating the encoder's states relative to each other. The Transformer's decoder takes in the context vector from the encoder *and* applies masked-attention to the existing output sequence, allowing the model to attend to previously generated tokens when generating the next token. The Transformer architecture has had a signif-

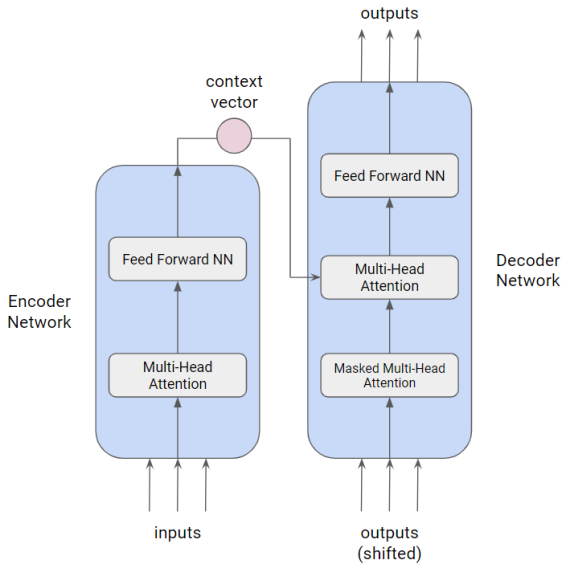


Figure 2.4 A basic visualization of a Transformer architecture. Note that this is an oversimplification.

icant impact on the field of NLP and has inspired many subsequent works that build on the architecture, including BERT (Devlin et al., 2018), ELMo (Peters et al., 2018), the

GPT family of models (Radford et al., 2018, 2019; Brown et al., 2020; Ouyang et al., 2022; OpenAI, 2022, 2023), and more. The subsequent sections will explain some key types of Transformer-based LLMs.

2.3.2 Transformer-Based LLMs

Transformer-based LLMs, also sometimes referred to as pre-trained language models, are all the rage in current NLP research.[9] These models reuse some or all of the Transformer architecture, particularly the self-attention and multi-head attention mechanisms. NLP researchers organize these models along many different axes, and can disagree on which models belong where. There exist differing opinions on how best to categorize LLMs, following from the large existing corpus of research and the increasing volume of papers published about LLMs each year. One possible categorization uses the model's training objective. This broadly splits Transformer-based LLMs groups, with two key categories: masked-language models, and auto-regressive models (Liu et al., 2023).

1. *Masked-language models* are LLMs that are trained to predict missing words in a sentence (Zhang et al., 2022). They are trained in a self-supervised manner by replacing a token, or word, in a training sentence with the [MASK] token. They learn the relationships between different words in a context, with the goal of predicting the most likely word or sequence of words to fill in the masked position. One famous masked-language model is `BERT` (Devlin et al., 2018), which achieved state-of-the-art results on a wide range of NLP tasks when it was released and has inspired a plethora of similar models.

2. *Auto-regressive LLMs* are deep learning models designed to model the probability distribution of a sequence of tokens, given the previous observations in the se-

[9] After this point, references to 'LLMs' are actually specifically to Transformer-based LLMs, unless otherwise stated.

quence (Zhang et al., 2022). In other words, their training goal is to predict the next word in a sequence, given all of the previous words. This enables the models to use its output as subsequent input when predicting the next in the sequence, updating it's probability vectors with the next context at each step. A prime example of an auto-regressive LLM is the GPT family of models, explained in detail in Section 2.3.5.

2.3.3 Prompting LLMs

With the dominance of general purpose LLMs, a new field of study in NLP has emerged: prompt engineering. LLMs receive natural text as input, transform it into a vector representation, and then do the reverse with their outputs. This natural text input is also referred to as a prompt. An intuitive explanation is to think of the LLM outputting y with the highest probability $P(x;\theta)$ where x is the prompt and θ is the model parameters. Prompting a general purpose LLM allows the model to conduct a variety of tasks without training directly for them.[10]

While a natural language prompt might be intuitively easy to create, in practice it can be hard to optimize, as small deviations can create large changes in the output (Liu et al., 2023). Plus, the frequency of new models and retrained model weights can mean that engineering an optimal prompt can be like trying to hit a moving target. Still, it is relevant to describe two basic types of prompts that relate to the training objectives seen in Section 2.3.2:

- A *cloze prompt* is one with a masked token for the LLM to predict, or fill (Liu et al., 2023). These are often seen with masked-language models due to their similarity with the model's training objective. For example, when conducting sentiment analysis, an prompt might be a movie review with the appended sen-

[10] N.b., This does not actually mean the model does not need to train heavily; as previously mentioned LLMs are trained on enormous amounts of data. It simply removes the need to train for a *specific* task.

tence "I [MASK] this movie.", where the model would predict a word related to the sentiment of the rest of the movie review.

- A *prefix prompt* is essentially a string prefix with optional input context (Liu et al., 2023). These are often used with auto-regressive LLMs, again due to their similarities with this model's training objective. For example, one might ask a LLM to "Write a story.", and then provide the first sentence of the story as context. The model would then predict the next likely works in the sequence, producing a story.

Further discussion of prompts and prompt engineering techniques can be seen in Section 2.3.6.

2.3.4 Text-to-Text Transfer Transformer Model (`T5`)

The Text-to-Text Transfer Transformer model, better known as `T5`, is a transformer-based auto-regressive LLM developed by Raffel et al. (2020). When it was introduced, `T5` achieved state-of-the-art performance on several NLP benchmarks, demonstrating its effectiveness and versatility. At the time, the unique aspect of the `T5` model was its 'text-to-text' approach, which is explained in the introduction of Section 2.3. This model is capable of many NLP tasks including summarization, translation, question answering, and classification; all of which are included as examples in the original `T5` paper. In this thesis, the `T5` model is used for paraphrasing (see Section 3.1 for further details).

2.3.5 Generative Pre-trained Transformer (GPT)

Generative pre-trained Transformers (GPTs) are a family of LLMs developed by OpenAI, based on the transformer architecture. The GPT family of models are auto-regressive models, so their training objective is to predict the next word in a sequence given the

previous words. OpenAI has trained multiple GPT models on a massive scale, building intractably complex models. The first GPT model, simply `GPT`, had 117 million parameters (Radford et al., 2018). `GPT-2` was increased to 1.5 billion parameters (Radford et al., 2019). `GPT-3` jumped up to 175 billion parameters (Brown et al., 2020). These three models work in very similar manners, with the key differences in performance stemming from the size of their training corpora and their number of parameters.

After `GPT-3`, OpenAI released a LLM with an important architecture change, called `GPT-3.5` or `InstructGPT` (Ouyang et al., 2022). This model was fine-tuned using what the authors call reinforcement learning with human feedback (RLHF). Using human-annotated data (i.e., the human feedback in RLHF) and supervised learning, a reward model is trained. This reward model computes a reward signal on a collection of outputs from `GPT-3`. The output with the highest reward is chosen, hence reinforcement learning is used to choose human-preferred outputs from `GPT-3`. Because of this additional layer, `GPT-3.5` hugely outperforms its predecessors in the GPT family, even when it has fewer parameters. OpenAI iteratively deploys this model, meaning querying it with OpenAI's API has regularly updated model weights.[11]

In November 2022, OpenAI released `ChatGPT`, which took the world by storm (OpenAI, 2022). They quickly followed up with the release of `GPT-4` (OpenAI, 2023). For proprietary reasons the full specifications of this model have not been released. ChatGPT is a sibling model to `InstructGPT`, meaning they are trained with the same architecture, but `ChatGPT` likely has different training data and a different number or parameters. As `InstructGPT` is the most advanced model with full specifications available at the time of this thesis, it is used for the experiments in Chapter 4.

[11] N.b., While improving performance, this creates reproducibility issues that are a problem in academia. To negate this in this thesis, the generations from OpenAI's API are included in the project's GitHub repository.

2.3.6 Controllable Text Generation

Controllable text generation (CTG) is a sub-task of the more general NLP task of text generation. 'Regular' text generation is when a model produces text in response to some input. For example, one might ask a LLM to vaguely "Write a story.". *Controllable* text generation is text generation conditioning on specific attributes, such as style, tone, sentiment, topic, or other (Liu et al., 2023; Zhang et al., 2022). For example, one might ask a LLM to "Write a story about a McGill University student." or "Write a story about a McGill University student from the perspective of their professor." or even "Write a whodunnit mystery about a McGill University student where their professor is the detective.". This is a challenging task in natural language processing, but it is becoming increasingly important as text generation applications become more widespread.

There are a large variety of approaches to achieve CTG. The following points focus on some approaches to achieve CTG with auto-regressive LLMs with prefix-style prompts, as these models and prompts are seen in the experiments conducted for this thesis. Other strategies for CTG include retraining LLMs, post-processing textual outputs, ensemble learning and more (Liu et al., 2023; Zhang et al., 2022).

Control Elements One approach is to condition the text generation process on certain attributes or features by adding them as a *control element* (Liu et al., 2023; Zhang et al., 2022).[12] For example, the model can be conditioned on a specific sentiment by way of adding a marker for sentiment inside a prefix-style prompt. This might look like changing the prompt from "Write a story." to "Write a happy story." With the increasing impressive ability of LLMs to generate text, even something as simple as this approach can have huge effects on the output text.

[12]Other works sometimes refer to these as *control codes*.

Fine-Tuning An alternative to using control elements is fine-tuning of a LLM. This involves additional training of an already pre-trained model on another, usually smaller, dataset that is specific to the task at hand (Liu et al., 2023; Zhang et al., 2022). For example, fine-tuning a LLM on a corpus of in a different language can improve results in machine translation. Unfortunately, fine-tuning a LLM can be very expensive with respect to time and compute resources (Liu et al., 2023; Zhang et al., 2022). An example of fine-tuning a LLM for educational question generation is touched on in Section 2.4.1.

Few-Shot Learning A simpler way to take steps towards the same idea as fine-tuning is called few-shot learning.[13] Rather than further training a model, examples of the desired output are provided to the model *within* the prompt. For instance, one might prompt a LLM with "Here is a recipe for chocolate cake: [insert recipe here]. Write me a recipe for a carrot cake.". Few-shot learning has been shown to adapt LLMs to unseen scenarios without additional training, making it a robust and exciting approach (Liu et al., 2023; Zhang et al., 2022). The reason this is called *few*-shot learning is that the word *few* represents the number of examples included in the prompt. Thus, zero-shot learning has no examples included in the prompt, one-shot learning has a single example included in the prompt, two-shot has two, and so on. Existing work shows that generally including more examples improves the generation results (Liu et al., 2023; Wang et al., 2022b), though there is some work that argues in favor of zero-shot learning being sufficient (Brown et al., 2020).

2.4 Question Generation

Question generation (QG) is the NLP task that aims to automatically generate valid and grammatically correct questions from a given context. The format of both the

[13] N.b., Few-shot learning can be used as a technique for a wide range of machine learning tasks, and can be defined slightly differently for these other uses. It is explained here within the context of this thesis and CTG.

inputs and outputs of the generation can vary wildly within the task: from simple short answer style questions, to multiple choice questions (that contain one or more correct options, as well as incorrect options); and from generating using a simple input sentence to generating textual questions from different modalities of data (Das et al., 2021). QG is often associated with the related task of question answering, where NLP models attempt to isolate the answer to a question from a context. Some QG models will try to generate the question and answer simultaneously, or generate a question while conditioning on the answer. The QG techniques focused on for this thesis are in a sense more 'traditional', meaning they are focusing on simple generation from context, without considering other data types, multiple data sources, or answer generation.

Question generation lends itself to an obvious use case in the generation of content for education. However, not all QG works are designed for this purpose. They might also be for applications in questions answering, information retrieval, and more. This section of the literature review will emphasize existing work concerned with educational question generation in particular, and highlight some survey papers and relevant works in this niche.

Over the course of NLP research, there have been many different approaches to generate questions, from early rule-based systems to today's powerful LLMs. Section 2.4.1 will explain a brief history of educational question generation. Until recently, this coincided quite closely with the research in the general case of QG. This is because many early papers are focused on demonstrating that QG is possible and that robust methods that can be applied to the educational setting exist; rather than performing QG with more specific pedagogical goals in mind. The majority of novel educational QG systems rely on deep learning techniques, specifically Transformer-based LLMs, to generate more nuanced pedagogical questions. This is the case for QG techniques used in this work. The reason for this is simply the success of such models on this task, and others in NLP (Dong et al., 2022; Lu and Lu, 2021). Thus, section 2.4.2 will give more

emphasis to recent works conducting educational question generation with LLMs.

2.4.1 A Brief History of Educational Question Generation

The history of educational question generation can be traced back to the early days of AI research, when researchers began exploring the use of computers to process and generate natural language. In the 1970s, researchers developed early QG systems that used rule-based approaches. These systems used hand-crafted rules and templates to generate questions from a given text, but they were limited in their ability to handle complex sentences and to generate questions that required deeper understanding of the text (Zhang et al., 2021). Rule-based approaches can be split into three key types:

1. Template-based approaches are generally robust but lack diversity in their generations. For example, it is simple to generate fill-in-the-blank questions by masking a word as seen in Agarwal and Mannem (2011). In this paper, the authors use a collection of features (e.g., word frequency, the height of the word in the syntactic tree, etc.) to optimize the selection of which key word to mask. Notably, the authors also generate a set of 'distractors' along with the question with a masked word for a student to choose from. While fill-in-the-blank questions are simplistic, they are still used in current applications such as in Van Campenhout et al. (2022), because these methods are easy to understand and implement, and they offer great reliability. Often template-based approaches are used in tandem with syntactic or semantic methods. This can be seen in some of the subsequent examples, where templates that take advantage of syntactic information can help rearrange declarative sentences into interrogative ones.

2. Syntax-based approaches use the underlying syntactic structure of their inputs to transform sentences into questions. The additional complexity offered by syntax-based approaches meant that more variety exists in the generations. One of the

very earliest examples of a syntax-based approach can be seen in work by Wolfe (1976). The author generated pedagogical questions by pattern-matching input sentences to a set of handwritten rules (i.e., a set of templates) and generating the rule's corresponding question. Their goal was to improve independent learning while reading, by way of asking students to answer questions while in the course of reading the material.

3. Semantic-based approaches use the underlying meaning of an input sentence, rather than its structure, to generate questions. For example, Yao and Zhang (2010) parse sentences into a minimal recursion semantic structure[14] that can then be realized as a question. Often these approaches are also used in tandem with syntactic information in order to reliably generate grammatically correct questions (Kurdi et al., 2020).

In the 2000s, with the advancement of machine learning and statistical approaches, researchers began applying these new methods to educational question generation. This included training models on large datasets of text and associated questions, using algorithms such as decision trees and neural networks to learn patterns and generate questions (Zhang et al., 2021). During this wave of research many NLP techniques were emerging to improve these models, such as tools to parse through syntax tree structures representing sentences. A combination of such NLP tools and statistical methods can be seen in work by Heilman and Smith (2010a), where the authors combine a ranking model with a syntax-based method to generate reading-comprehension questions. First, they use a syntax tree representation of an input sentence to perform sentence simplification. Next, they apply a set of syntax rules to transform a sentence into a question phrase. Finally, they use an overgenerate-and-rank approach where many candidates are generated and a logistic regression model is used to select

[14]Minimal recursion semantics are a meta-level semantic representation of a sentence. Read more about this structure in Yao and Zhang (2010).

the best candidate questions from a larger set. While still relying on some rule-based methods, works such as this were able to generate a more diverse set of questions by utilizing new NLP and ML technologies. Novel statistical approaches are dominated by Transformer-based methods which covered in detail in the subsequent section.

2.4.2 Educational Question Generation with LLMs

The use of Transformer-based models for educational question generation has far outpaced the success of previous approaches to QG (Kurdi et al., 2020; Dong et al., 2022; Lu and Lu, 2021). These models have shown very promising results in generating questions that are more diverse and accurate, and in handling complex examples. For example, Du et al. (2017) used an attention-based sequence learning model to generate reading comprehension questions. They compare their approach with sentence and paragraph level inputs, but find that they have greater success when only incorporating sentence-level information. Since then, more recent works using LLMs have had success incorporating additional context (Dong et al., 2022; Lu and Lu, 2021). Accordingly, Transformer-based LLMs methods are at the forefront of current research into both general purpose and educational question generation.

Many works focus on developing systems and models to optimize the generation of pedagogical questions. Unfortunately, there is often a reliance on automatic metrics to assess generated outputs, despite the fact that these metrics have been shown to be insufficient both inside the educational domain and beyond (Laban et al., 2022; Kurdi et al., 2020; Das et al., 2021). Alternative evaluations involve human ratings, which can be resource intensive, but are still important. For example, Heilman and Smith (2010b) conduct a human-evaluation of automatically generated reading-comprehension questions, which are used to train the aforementioned ranking model used in Heilman and Smith (2010a).

Reliance on automatic metrics can be in part attributed to the fact that educational

QG research has very few cases of real world deployment to show (Kasneci et al., 2023; Kurdi et al., 2020). The reason for this has been explored in a need-finding study by Wang et al. (2022a). This paper aims to bridge the gap between the huge educational potential of QG systems and the lack of adoption of these systems in real world classroom settings. The authors call for QG systems to be modular (i.e., consisting of a collection of smaller tasks), process-oriented (i.e., iterative and fluid, matching the process of teachers writing questions), and handle diverse input sources. In order for educational QG to be more widely deployed, the question generation needs to meet the needs of teachers and students alike. Chapter 4 takes steps to show that personalization of educational question generation can meet these needs.

Despite minimal classroom deployment, there are still some informative works that perform QG experiments with real teachers and students, such as:

- Laban et al. (2022) design a quiz writing task to attempt to get teachers to create educational content with the help of QG. During this task, teachers would choose from a list of topics and make a quiz with candidate questions that were automatically generated, marking the candidates as acceptable or not as they went. In the whole study, 3,164 questions were annotated with a global acceptance rate of only 52%.[15] This experiment is informative as to why teachers might be hesitant to use automatically generated questions, and that more advanced models are needed to generate viable questions.

- Van Campenhout et al. (2022) explain an NLP system for translating textbooks into interactive course ware. The authors report on the results of their generated material with a large group of students, constituting a huge user study with automatically generated questions. Their results show that students perform just as well on their machine-generated questions as on human-written ones. Un-

[15]This paper used a variety of SOTA models at the time, up to and including GPT-2. More recent models are likely to improve this rating, as per their successes elsewhere.

fortunately, their QG system is mostly rule-based, as their generated questions are concept-matching and fill-in-the-blank style. This leaves remaining questions about the success of more diverse automatically generated questions.

- Given the current excitement surrounding LLMs, there is a plethora of opinion papers and blog posts of teachers saying they are already using LLMs for the generation of their educational content (Baidoo-Anu and Owusu Ansah, 2023; Terwiesch, 2023).

Beyond the question of real-world deployment, much of the current literature on educational QG does not focus on personalization of generated candidates (Kurdi et al., 2020). This personalization includes generating questions at specific difficulty levels and forms that are enriching for individual students. This under-explored research area in the field is a focal point of Chapter 4. That being said, there are still some precedent works to note. Srivastava and Goodman (2021) fine-tune a LLM to generate questions adapted to particular difficulty levels of students learning a second language. Essentially, they model the difficulty level suitable for students given their previous history, and use this as input to a QG model. Their work involves a huge corpus of student data, but no direct interaction with real students or teachers. Chinkina and Meurers (2017) generate different types of fill-in-the-blank questions in order to target different learning goals for second language learners. They conduct a human-evaluation study to evaluate their generations by comparing a corpus of human-authored questions to their automatically generated ones. However, this study uses a crowd-sourcing platform with English speaking participants, who are not necessarily teachers or students. Wang et al. (2022b) use CTG with LLMs to generate educational questions of different types. They try a combination of different prompting strategies and conclude that shorter input contexts and few-shot learning (with examples from related topics to the input context) improve the quality of candidates generated with LLMs. Similar results are seen in Section 4.2, though the models used and prompt engineering of the CTG is

different. The authors of this work also suggest that evaluating such generations with real teachers is a required follow up step to their work, which is seen in Section 4.3.

Chapter 3

Personalizing the Learning Experience

Intelligent tutoring systems (ITS) are AI-based computer systems capable of automated teaching. As explained in Section 2.2, they have the potential to provide accessible and scalable education to students around the world. Combining ITS with the studies mentioned in Section 2.1, which show that students learn significantly better in one-on-one tutoring settings than in classroom settings leads to the hypothesis that ITS with integrated personalization for their users will outperform their static counterparts and increase the benefits that students receive. It has been shown that personalization can be addressed in an AI-driven, dialogue-based ITS, and can have significant impact on the learning process (Kochmar et al., 2020). Personalized learning experiences can be created in different ways, from dialogue feedback to student-specific question selection (St-Hilaire et al., 2022). To the best of our knowledge the published work presented in this chapter was the first exploration of personalization in question *phrasing*.

Chapter 3 addresses the personalization of question phrasing for students interacting with an ITS. In other words, the way a question is written can have huge effects on student understanding, and as such is an excellent target for personalization. Moreover, the potential impact of fitting linguistic realizations of questions to the needs of a student is massive, as questions are one of the main pedagogical intervention types

used by teachers in general and in dialogue-based ITS. Pedagogical research shows that students benefit from being asked questions tailored to their level of subject expertise and their needs during in-person tutoring sessions (Ashton-Jones, 1988; Hrastinski et al., 2021). The key hypothesis researched in this chapter is whether the same effect can be achieved when questions are adapted to the students' levels of expertise within an ITS. To test this hypothesis, question variants created by a human domain expert were integrated onto the Korbit platform and an A/B test was run. As previously mentioned, Korbit's AI tutor, `Korbi`, is a dialogue-based ITS. The question variants themselves are discussed in Section 3.1. Section 3.2 explains the selection process for which question variant should be matched to a student, based on their history on the platform. The A/B test run with `Korbi` is covered in Section 3.3. The results from this A/B test support the hypothesis due to the increased success seen with students who receive questions which have been matched to them.

3.1 Handwritten Question Variants

A set of 180 question variants were handcrafted by a human domain expert from 60 existing questions on the Korbit platform. The variants were designed to retain the same meaning and acceptable solutions as the original question, while reflecting a question taxonomy with three levels of difficulty: *beginner*, *intermediate*, and *advanced*.[1] Figure 3.1 contains an example question and it's related variants. In order to attain variants in each of the three levels, there were some assumptions made about how question phrasing affects difficulty. Prior research in both the pedagogical field and the application of question generation to educational contexts shows that less detailed questions are harder (as the student must have more background knowledge in order to understand and answer the question), and more elaborate questions are easier (as they 'hint'

[1]See Section 2.1.3 for further explanation of level-difficulty question taxonomies.

at the answer with extra information) (Taylor, 1962; Kurdi et al., 2020). It is important to note that while these general trends have been shown to hold true, there are edge cases of students who will not experience less detailed questions as harder, and more detailed questions as easier. Moreover, a question can very well be conceptually difficult, regardless of a particular linguistic interpretation of it. A more robust system would map between many different kinds of question variants and student profiles. However, this complex mapping is out of scope for this chapter. Instead, the goal here is to prove that personalization of question phrasing can benefit students in an ITS, not optimize the benefits.

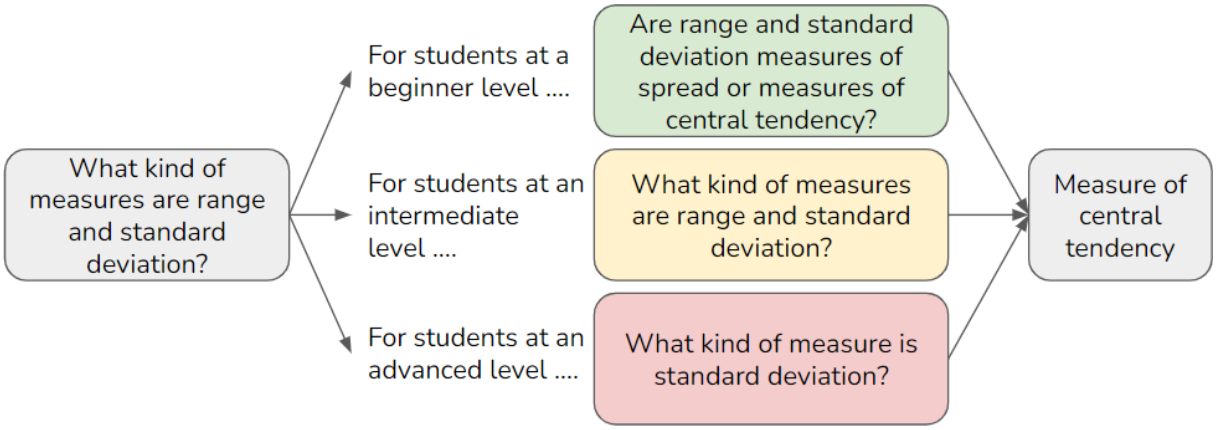


Figure 3.1 An exemplary question adapted to different difficulty levels while retaining the same correct answer/group of correct answers (the different pluralities is small enough to be overlooked).

In the final variant set, each question has three variants at different levels of proficiency. These variants were made easier by adding elaborations and synonym replacement, and more difficult by removing non-essential explanations and synonym replacement. As the mean word count column in Table 3.1 shows, the beginner variants are longer and the advanced ones are more concise. The variant set was given to three human experts who are English-speaking educational content creators with at least an MSc in a related field. The annotators rated them on three ordinal scales from

0 to 5, defined as follows:

- *Difficulty*: The relative complexity of the question as compared to the other variants. A score of 0 implies a trivially easy question, and a score of 5 implies an advanced question that some students would be unable to answer.
- *Fluency*: The correctness of the spelling, syntax, and grammar. A score of 0 implies an unreadable/not understandable question, and a score of 5 implies a perfectly fluent question.
- *Meaning preservation*: The preservation of the meaning and desired answer between the original question and the variant. A score of 0 implies a completely unrelated question, and a score of 5 implies an exact match between both version's expected (or acceptable) answers.

Table 3.1 Mean variant scores from human experts, and average word counts by level. Arrows indicate better scores for strictly directional metrics. *Difficulty*, *Fluency*, and *Meaning Preservation* are on a scale from 0 to 5.

Level	Difficulty	Fluency (↑)	Meaning Preservation (↑)	Word Count
Beginner	1.689 ± 0.635	4.600 ± 0.471	4.789 ± 0.451	39.800
Intermediate	2.667 ± 0.689	4.683 ± 0.481	4.839 ± 0.406	33.533
Advanced	3.939 ± 1.269	4.544 ± 0.661	4.717 ± 0.516	27.433

The results of their annotation can be seen in Table 3.1. The *fluency* and *meaning preservation* metrics are consistently high across all difficulty levels. The *difficulty* metric increases with the assigned levels, as expected. The average Spearman correlation coefficient across experts and metrics is 0.41, which is acceptable as it can be interpreted as 'moderate', 'fair', or even 'strong' depending on the interpretation scale used (Akoglu (2018)). As a result of the annotation, a few questions underwent slight rephrasing when the annotators agreed that a change would be beneficial. No questions received poor enough scores to be discarded, as evidenced by the high mean results.

3.2 Choosing Question Variants for Students

In order to assess if the level-adapted variants benefit student learning, it is necessary to map sufficiently well between the variants and an individual student's needs. In other words, to learn to select an appropriate question variant for each student at each step in the dialogue. Previous works in educational QG provide precedent to use a student's history (i.e., previous success/failure) to discern how difficult of a question they should receive (Srivastava and Goodman, 2021).

A dataset of anonymized student history from the Korbit platform was collected for the purpose of learning this mapping. It contains 2,137 student sessions with the platform. Each student's session history consists of all of the exercises they encountered and their attempts to solve them. The student attempts include information indicating if the student succeed on the exercise, failed and received feedback, or if they chose to skip the exercise altogether. Each exercise attempt (except the final one in a given session, as there is no next exercise to act as the target value) was included as a point in the dataset, for a total of 13,504 data points. The target variable of this dataset is the student's success on the *next* exercise attempt they submit. With this data it is possible to calculate a set of heuristic features indicative of a student's level, and subsequently build a logistic regression model to predict if a student will succeed on the next exercise.

Seven features were calculated from the collected dataset whose definitions are shown in Table 3.2. In order to select an optimal set of these features to predict the target variable, a grid search was performed. All possible combinations of these features were used to train a logistic regression model and tested on a 30% held out test set. The models' performances were compared on the basis of accuracy. From the original set of seven features, two features, *topic success* and *topic skip*, were chosen. The selection was made in order to balance between the need for accurate predictions and the need to have a small number of features to calculate (to minimize the time require-

ments of selecting a variant on a live platform). Using just these two features the final model is able to predict if a student will get the next exercise correct with an accuracy of 80%. Given the results demonstrated in Section 3.3, this accuracy is sufficient to provide significant learning gains to students. A more complex and accurate model of a student's level would likely improve the results seen even further (for example, this might be achieved by building on work by Srivastava and Goodman (2021)).

Table 3.2 Features considered in next-exercise-success prediction model. N.b., a topic on the Korbit platform is a broad category of material, such as 'Probability' or 'Deep Learning'.

Feature Name	**Definition**
Topic Success	A numerical feature in $[0, 1]$ that is the eventual success rate per all exercises previously attempted in a given topic.
Topic Skip	A numerical feature in $[0, 1]$ that is the skip rate per all exercises previously attempted in a given topic.
Recent Topic Success	A numerical feature in $[0, 1]$ that is the eventual success rate on the last 10 topic exercises.
Recent Topic Skip	A numerical feature in $[0, 1]$ that is the skip rate on the last 10 topic exercises.
Recent Success	A numerical feature in $[0, 1]$ that is the eventual success rate per the last 10 exercises attempted (regardless of topic).
Improvement	A binary feature that is 1 if the users success rate on the most recent 5 exercises is greater than previous 5, and 0 otherwise.
Recent Attempt Count	A numerical feature that is the average number of attempts on the 10 previous exercises.

The outputs of the next-exercise-success prediction were used on the live Korbit platform experiments to assign variants to students given their history. Essentially, the aforementioned two features were re-calculated each time a student got a new exercise. Then, the pre-trained logistic regression model generated a probability that the student will succeed on the new exercise. Students were then assigned difficulty-level variants based on which third of the percentile range that their probability of success falls into. These percentiles are also pre-calculated from the student history dataset. In other words, students in the 0^{th} to 33^{rd} percentiles get beginner variants, students in the 33^{rd}

to 66^{th} percentiles get intermediate variants, and the students in the 66^{th} to 100^{th} get advanced variants.

3.3 A/B Test

Methodology To test the question phrasing personalization hypothesis, the variants and variant assignment strategy described in Section 3.2 went live on the Korbit platform to interact with real students. The experiment was an A/B test that ran for over two months, collecting data from over 400 students at varied skill and experience levels.[2] Student attempts were divided into three groups. The *expected* variant group received the variant which the variant assignment strategy suggested to them. The *non-expected* variant group received a variant which was *not* the variant assignment strategy suggested to them (e.g., a beginner question variant for an advanced student). The *control* group students received the original variant regardless of their history (i.e., that which was already on the platform before this experiment).

Three key metrics were recorded through the course of the A/B test. Firstly, *solution acceptance rate* is the proportion of successful attempts per total exercise attempts. However, as explained in Section 2.1.2, research (and intuition) demonstrates that simply succeeding on exercises does not equate to learning. Students should instead be challenged within their zone of proximal development but eventually obtain the right answer (Hedegaard, 2012). Thus, it is important to minimize the *ultimate failure rate* as opposed to simply maximizing attempt success, where *ultimate failure rate* is the proportion of failure out of all exercises seen by students. Unlike *solution acceptance rate* which shows the success rate per **attempt**, *ultimate failure rate* shows the fail rate per **exercise**. Finally, the *skip rate* is also measured, which is the number of exercises a stu-

[2] An A/B test is a controlled experiment in which online users are assigned to one of two or more experimental groups (one of which is usually a control group) and their experiences on a platform are compared (Sammut and Webb, 2017).

dent skips out of the total number of exercises they encounter. This metric is indicative of a student's engagement. Intuitively, the more they skip, the less they engage with the content.

Ethical Considerations As with all experiments involving human participants, this study has a few ethical considerations worth noting. Firstly, McGill's Ethics Review Board reviewed and approved an application which encompassed the work seen in this A/B test. Secondly, all students who interacted with the A/B test signed an informed consent form to participate in the experiment upon signing up to Korbit's website. These participants signed up to Korbit of their own accord, usually finding out about the platform through social media or word-of-mouth. There was no compensation offered to them for their participation, barring their access to Korbit's educational resources. Similarly, there was no penalty or reward provided for their failure or success on exercises. As the participants are not required to disclose any information about themselves other than a name and email address when they sign up for Korbit, we cannot comment on the demographics of the participant pool. It is possible that this is a source of bias, as a majority of participants could be coming from similar demographics.

Results and Analysis All three of the metrics show the *expected* group performing the best, followed by *control* and finally *non-expected*. For *solution acceptance rate* and *ultimate failure rate*, the difference between *expected* and *non-expected* groups is statistically significant at $\alpha = 0.05$ by a Student's t-test.

Across all metrics, the improvement between the *control* group and *expected* group indicates that the addition of personalized variants is helpful for student's learning. The statistically significant difference between the *expected* group's and *non expected* group's results indicates that fitting a question variant to a student's level will improve their success, where a mismatch can be detrimental to student's learning. Thus,

Table 3.3 Test results. Arrows indicate better scores for strictly directional metrics. Metrics marked with * have a statistically significant difference between them at the $\alpha = 0.05$ level by a Student's t-test.

Experiment Group	Solution Acceptance Rate*	Ultimate Failure Rate* (↓)	Skip Rate (↓)	n
Expected	0.626 ± 0.069	0.163 ± 0.053	0.105 ± 0.044	190
Non-Expected	0.468 ± 0.083	0.295 ± 0.076	0.144 ± 0.058	139
Control	0.596 ± 0.081	0.191 ± 0.065	0.121 ± 0.054	141

personalization is a great tool to increase learning gains but must be done properly to achieve the desired results.

The difference between the *expected* group and *control* group is smaller than the difference between the *expected* group and *non-expected* group. This can be attributed to the fact that the original questions were refined through several rounds of review by domain experts when they were created for the Korbit platform, whereas the variants only were reviewed once. Additionally, the *control* group's exercises are always at an intermediate or advanced level, while the strongest results in support of question variants are seen with beginners. Isolating the students who score for beginner variants only, we see a 19% relative reduction in *ultimate failure* when comparing the *expected* to the *control* group, which demonstrates a bigger impact for beginners.[3] This is substantially larger than the difference seen in Table 3.3, which demonstrates that beginner students are performing much better when variants are matched to their level. Additionally, the same comparison shows a 30% relative reduction in the *skip rate*, suggesting that the beginners are more engaged when dealing with beginner variants.

[3]Relative reduction is a comparison of the metric values in two groups calculated as $V_a - V_b/V_a$, where V_a is the value in the first group, and V_b is the value in the second group. It is commonly used in epidemiology (Porta, 2016).

3.4 Conclusion

In this chapter, an experiment designed to explore whether the positive effects of personalization seen in one-on-one tutoring can be replicated within an ITS was explained. The experiment involved an A/B test where students on an ITS either received question variants matched to their level (the *expected* group), random question variants (the *non-expected* group), or the original versions of the questions (the *control* group). The results showcases a clear improvement in the success of students in the *expected* group over the other two groups. Thus, the hypothesis that providing question variants suited to student's level will improve their learning gains is confirmed by the A/B test. In particular, personalization of question phrasing appears to be useful for beginner level students who need more assistance, which is an encouraging and intuitive result.

Chapter 4

Generating Educational Questions

Chapter 3 demonstrated that generating variants from existing questions has great pedagogical value. An immediate next step for the work in Chapter 3 is to automatically generate these variants. This can remove, or at least decrease, the additional workload placed on teachers who may wish to personalize their content in order to benefit their pupils. Adapting existing questions to variant versions can be done in a variety of ways. To this end, three approaches are outlined in the Section 4.1, as well as corresponding small-scale human evaluations to assess the generated candidates' quality.

The generation of question variants requires a pre-existing high quality set of questions for use. There are many cases where such a set of questions is not readily available. Accordingly, Section 4.2 focuses on generating educational questions from scratch; using only educational text as input and a large language model. It is clear to see how combining robust pedagogical question generation with question variant generation might open the floor to virtually endless personalized exercises. Before claiming this possibility, it is vital to ensure that any generated content is high quality and pedagogically useful. As such, Section 4.3 outlines a human annotation experiment whose goal is to show the pedagogical value of the candidates generated in the previous sec-

tion. The results of this experiment demonstrate that across the board teachers find the generated questions to be of high enough quality to be sufficiently useful for their own classrooms. In sum, these various QG methods can serve as tools for teachers to increase the efficiency of their content creation process; and help them to provide a better learning experience to their students.

4.1 Automatically Generating Question Variants

Chapter 3 demonstrated that personalization is a valuable tool to improve the learning gains of students. However, the extra effort required to produce high quality questions and related variants exponentially grows with the more difficulty levels, topics, and other axes one considers. To make the value of personalization realistically attainable, there needs to be some way to automate the creation of pedagogical questions and their variants. This section deals with the generation of variants for existing questions. This can be done with a variety of approaches, only a few of which are outlined in this thesis. With the increasing capabilities of LLMs, applying newer models to variant generation would likely lead to even better variants, and in turn improved student learning experiences.

Each variant generation approach presented underwent a small human evaluation to validate its potential. For all three methods, the question set used is pulled from the educational content on the Korbit platform (the ITS introduced in Section 2.2). This content includes over 1400 questions written in natural language on the topics of data science, machine learning, artificial intelligence, mathematics, and statistics. Additionally, there are some specific projects included in the content that apply the concepts of these fields to different domains, such as finance. Question types within this set include short answer, long answer, multiple choice, numerical answers (e.g., math equations and/or their solutions), and more. Examples of the original questions and their corre-

sponding variants can be found within the subsequent sections, as well as in Appendix A.

Due to resource constraints, the variant experiment with real students found in Chapter 3 was not replicated with automatically generated variants. However, by demonstrating the success of automatic variant generation it is clear to see they could be a good replacement of human-written question variants used there. Future work will need to tackle the problem of learning to map from these readily available variants to the different question levels in order to apply the variant selection approach outlined in this experiment.

4.1.1 Sentence Simplification

Applying automatic sentence simplification strategies to educational questions has the potential to generate lingustically simplified versions of the original queries. If the originals are altered to use more common words, or less complex sentence structures, then these simpler variants can be intuitively easier for students to comprehend. Simpler variants have the potential to make educational questions more accessible to a wider range of people. For example, simplification could be especially helpful to non-native speakers who might possess a smaller vocabulary.

AudienCe-CEntric Sentence Simplification (ACCESS) is a recent sentence simplification model by Martin et al. (2019). This model is a transformer based sequence-to-sequence model that has a built-in control mechanism. Users are able to set the value of four attributes to control the simplification: *length*, *amount of paraphrasing*, *lexical complexity*, and *syntactic complexity*. The authors of this model also perform hyperparameter tuning to find the optimal values for these four control attributes that generated state of the art results on the WikiLarge test set at the time of submission.

ACCESS was used to assess if sentence simplification is a good fit for question variant generation. A vanilla version of the model, with the optimal hyperparameters men-

tioned previously was used to generate a set of question variants from the question set explained at the beginning of Section 4.1. A small human evaluation of 40 randomly selected simplified candidates was conducted by three annotators. These annotators are employees at Korbit, meaning they're familiar with the content and structure of educational questions, and have a confident grasp of both the material and the English language. The following three metrics were annotated:

- *Simplification*: A metric denoting if there has actually been a simplification between the original and the resulting question (if the model is not confident enough in it's simplification, it makes no changes). Since the metric definition can be rephrased as a 'yes or no' question, this metric is binary. An ideal mean value of 1 means that all of the questions in the set were simplified.

- *Correctness*: An ordinal metric from 0 to 5 denoting if the simplified question is grammatically correct; where a 0 score is unreadable/not understandable due to grammar mistakes or invalid synonym substitution and a 5 score is defined as perfect simplified sentence which is clear and valid. The scale is included instead of a binary metric since the severity of a grammatical error can vary wildly, and minor errors (e.g. forgetting to capitalize) are not as important to consider. An ideal mean value of 5 means that all of the questions in the set are grammatically correct.

- *Meaning preservation*: A metric denoting if the simplified metric is asking the same question as the original; in other words, if the new linguistic realization will be interpreted in a way that will lead to the same answer. Since the metric definition can be rephrased as a 'yes or no' question, this metric is binary. An ideal mean value of 1 means that all of the questions in the set preserve the meaning.

The simplifications done by `ACCESS` are relatively minor, but still powerful. For example, one of the questions fed to the model was:

*What is the purpose of **converting** to a standard normal distribution?*

Already, this question is reasonably simple. As such, ACCESS only made a small change, simplifying this sentence to:

*What is the purpose of **changing** to a standard normal distribution?*

The small word change may seem unimportant, but for a non-native speaker the more common diction might be just enough to make this question accessible. Additional examples can be found in the Appendix A where ACCESS performs similar word changes, or splits longer sentences into parts, increasing a question's readability. There are also examples of where the simplification fails, either by not making changes or by making erroneous changes.

Table 4.1 ACCESS simplified question variant annotated results. Arrows indicate better scores for strictly directional metrics.

	Simplification (↑)	**Correctness** (↑)	**Meaning Preservation** (↑)
Mean	0.949	3.730	0.784
Standard Deviation	0.226	1.111	0.866

The annotation results of ACCESS's simplified educational questions can be seen in Table 4.1. Each of the three annotators rated all 40 question variants, and the scores presented are the mean of their ratings. The results are encouraging, with 95% of the candidates showcasing a simplification, and 78% of the candidates preserving the meaning of the original. Additionally, the average *correctness* for the candidates is 3.7 out of 5, which is relatively high. With the addition of some automated grammar checking tools or other filtering mechanisms, these scored could easily be increased. Overall, these results show that the ACCESS model is capable of generated simplified question variants that have great potential to be used for educational content personalization.

4.1.2 Elaboration Generation

In contrast to simplifying questions to form variants, it is possible to expand the questions to include extra helpful details. This additional text can serve to help a student recall information related to the answer they have been asked for. This can help many different students, such as those new to a certain concept or those reviewing topics learned long ago.

One intuitive way to incorporate additional information is to include the definition of a key complex term contained within the question. The first step to generate such variants is to isolate a relevant keyword. This was done automatically using the complex word identification (CWI) strategy from Gooding and Kochmar (2019). In brief, as opposed to simply choosing the word with the maximum term-frequency inverse document frequency (TF-IDF), their strategy uses a sequence labeling model. This allows for the inclusion of the context around a word when searching for the most complex words in a text. The authors argue that context influences the actual complexity of a word, and should be considered in the CWI task. At the time of release, their approach had state of the art results. This CWI strategy was applied to the aforementioned set of Korbit questions to isolate their most complex words. The chosen word was cross referenced with a glossary of machine learning and data science terms compiled by the author and other employees at Korbit. This process changed a question such as:

What is the purpose of converting to a standard normal distribution?

into the following version with additional hinting information:

What is the purpose of converting to a standard normal distribution?
Standard Normal Distribution *= A normal distribution with a mean of 0 and a standard deviation of 1, written* $Z \sim N(0, 1)$ *.*

This added definition might help a student recall what a standard normal distribution is, note the specific mean and standard deviation, and prompt them to remember

where it might be useful. Unfortunately, this kind of addition can sometimes render a question useless. For instance, the original question:

What does the threshold value represent?

becomes trivial with the following addition of a definition:

What does the threshold value represent?

Threshold *= A value beyond which there is a change in the manner an algorithm proceeds.*

Additional examples of where the elaborations succeed and fail can be found in Appendix A.

As before, a small human evaluation with the same three annotators was conducted for 38 of these elaborated variants. Since these definition additions do not edit the original question, only add to it, they cannot introduce *correctness* or *meaning preservation* issues, as they were previously defined. Therefore, these variants were evaluated according to two different metrics:

- *Keyword Choice*: A binary metric denoting if the defined keyword is the optimal choice (in other words, if the annotator would have chosen the same keyword to define as a hint). Since the metric definition can be rephrased as a 'yes or no' question, this metric is binary. An ideal mean value of 1 means that all of the questions in the set had their best keyword selected.

- *Trivialization*: A binary metric denoting if the addition has made the exercise trivial. For instance, this could happen if the original question was asking for the definition of the most complex word. Since the metric definition can be rephrased as a 'yes or no' question, this metric is binary. An ideal mean value of 0 means that none of the questions in the set were trivialized.

The results of this human annotation can be seen in Table 4.2. As before, all three annotators saw all of the question variants, and the scores presented are the means

Table 4.2 Elaborated question variant annotation results. Arrows indicate better scores for strictly directional metrics.

	Keyword Choice (↑)	**Trivialization** (↓)
Mean	0.9324	0.0811
Standard Deviation	0.2527	0.2748

of their ratings. The annotators agree with the CWI model's selected keyword 93% of the time. And the questions are only trivialized by their additions 8% of the time; though this is likely to increase if the set of existing questions has a higher percentage of definition-type questions. The generation of these elaboration question variants is clearly works well. However, despite a reasonable intuition about the helpfulness of these variants, more research is required to assess if these added definitions actually benefit students requiring easier variants.

4.1.3 Paraphrasing

Models for paraphrasing are an obvious way to generate variants of a question. Section 2.3 explained the `T5` encoder-decoder model introduced by Raffel et al. (2020). There exists a fine-tuned version of `T5` specifically for paraphrasing called the `Parrot` paraphraser model by Damodaran (2021). `Parrot` was originally intended as an augmentation framework for NLU tasks, but its paraphrasing capabilities can be applied to many other tasks. This model was fine-tuned on a collection of paraphrasing datasets, including MSRP Paraphrase, Google PAWS, and Quora question pairs.[1] The model allows for fine grain control on three attributes: *adequacy*, *fluency*, and *diversity*. These attributes can be set to adjust the outcomes of paraphrasing, similar to the control knobs of the sentence simplification model explained in Section 4.1.1. Experiments in this thesis use the default values for these parameters.

[1]The author states that some of these were used to fine-tune the model but does not clarify which, presumably for proprietary reasons.

The resulting paraphrased variants are sometimes phrased completely differently than the originals. For instance, the following original exercise about aging receivables is written as a statement:

Explain the motivation of prioritizing aging receivables in the context of evaluating late customers.

The adapted version by `Parrot` changes the exercise into a question:

Why are aging receivables prioritized in the context of evaluating late customers?

Rephrasing a statement into a question presents the query at hand in a new way which can fit better with a student's needs or a teacher's goals. The paraphrased variants can also be condensed versions of the original. For example, observe the following original question about accessing values in a Pandas series:

What approach can be used to get the first 5 rows of any Series where it exists?

This question is adapted to a shortened version of the same:

What is the easiest way to get the first 5 rows of a series?

Shortened questions might leave less room for ambiguity and speed up the question understanding for a student (i.e., what does 'where it exists' mean here?). It is not obvious weather these variants will be easier or harder than the originals, and they will likey vary on a case-by-case basis. More examples of paraphrased outputs from `Parrot` can be found in Appendix A.

Table 4.3 `Parrot` paraphrased question variant annotation results. Arrows indicate better scores for strictly directional metrics.

	Correctness (↑)	**Meaning Preservation** (↑)
Mean	4.503	0.713
Standard Deviation	0.747	0.453

To assess a larger set of the variants generated by paraphrasing questions, the author evaluated 750 variants. The metrics seen in Section 4.1.1, *correctness* and *meaning preservation*, were assessed. Paraphrasing does not necessarily have the goal of simplifying a question; so the *simplification* metric was left out. The results of this assessment can be seen in Table 4.3. They show that the paraphrased variants are nearly always grammatically correct, with an average *correctness* of 4.5 out of 5. Additionally, the paraphrases are reasonably capable of maintaining the original question's content, with a *meaning preservation* of 71%. These results show that a simple pass of an out-of-the-box paraphrasing model is capable of generating viable question variants for personalization of educational content. Future work is necessary to map between paraphrases and individual student needs.

4.2 Preliminary Experimentation for QG with LLMs

While generating variants of questions is an exciting path to personalization of pedagogical content, this approach makes the assumption that there already exists a set of high quality educational questions. Consequently, this section of Chapter 4 is devoted to generating educational questions directly from existing natural language text on a given topic, such as textbook material or Wikipedia articles. Recent advances in the controllable text generation (CTG) abilities of LLMs, explained in Section 2.3, make this a promising direction for educational content generation.

In order to assess the applicability of LLMs and CTG to automatically generating educational questions, a preliminary experiment and qualitative assessment were conducted using a cutting edge LLM, `InstructGPT`. The goal was to determine the best set of experimental settings to generate a diverse and high quality set of questions that have the potential to be useful in a classroom setting. A helpful intuition for this preliminary experiment is that of grid search from hyperparameter optimization. In grid

search, every possible combination of parameters is attempted and compared to isolate the optimal subset. Similarly, this experiment compares the generation parameters of *context length*, *context domain*, *shot-setting*, and *control elements* to find the best performing parameter set. Each of these parameters will be explained in Sections 4.2.2 to 4.2.5, along with their results in the qualitative assessment.

4.2.1 Generation and Assessment Procedure

Contexts Candidate educational questions were generated from 45 input contexts. These contexts were gathered manually from Wikipedia and pre-processed. Their lengths and domains are discussed in Sections 4.2.2 and 4.2.3, respectively. Pre-processing included removal of citations, hyperlinks and phonetic spellings; formatting of full sentence bullet-point lists into paragraphs; and other minor data cleaning steps. The specific Wikipedia articles chosen were generally hyperlinks from the domain's main Wikipedia page (otherwise, from the domain's glossary page) in order to ensure they were relevant to the foundations of the given domain. Examples of these contexts can be found in Appendix B.2.

Generation Setup The questions evaluated in this experiment were generated using `InstructGPT` accessed through OpenAI's API. The prompt template in Figure 4.1 was filled in with one of the aforementioned input contexts, a question type *control element*, and between 0 and 5 few-shot examples. The nature of both *control elements* and the few-shot examples is explained in subsequent sections. Every possible combination of parameter settings was used to fill in the prompt and query `InstructGPT` resulting in 6423 generated candidate questions.

Qualitative Assessment Setup and Metrics A qualitative assessment was conducted by way of the annotation of a sample of the generations. The goal was to reach a pre-

```
Generate {control element} questions.

Passage: {example_context}
Question: {example_question}

Passage: {context}
Question:
```

Figure 4.1 Generation Prompt Template (*one-shot* template)

liminary understanding of their quality, as well as to learn which set of experimental parameters optimized the educational QG. As this experiment was only preliminary, all annotations in Section 4.2 are conducted by the author. The qualitative metrics assessed are as follows:

- *Relevance*: A binary variable representing if the question is related to the context provided. In order for a question to be on topic, at least one key concept from the context passage must be referenced or mentioned in the candidate. Note that the context doesn't necessarily have to contain the answer to the question on this concept (see the *answerability* metric). An ideal mean value of 1 means that all of the generated questions are relevant.

- *Grammar*: A binary variable representing if the question is grammatically correct. Any grammatical error (including capitalization or other minor errors) results in an ungrammatical question. An ideal mean value of 1 means that all of the generated questions are grammatically correct.

- *Adherence*: A binary variable representing if the question is an instance of the question type provided by the *control element*. This is done with the definitions of

question types at the discretion of the annotator. An ideal mean value of 1 means that all of the generated questions adhere to their goal question type.

- *Answerability*: A binary variable representing if there is a text span from the context that answers the question, or that could lead to an answer with no other information required (e.g., a student's opinion about presented facts could be answerable). Note that any reasonable answer is acceptable, it does not have to be the best/most complete answer to the question. An ideal mean value of 1 means that all of the generated questions are answerable.

Automatic metrics were also calculated across the full set of generations. However, these results were not as informative as the qualitative assessment, and have been excluded for brevity.

Qualitative Assessment Summary Results The assessment was conducted on a stratified random sample of this large generated set, in order balance the time constraints of a single annotator with the need to have a sufficiently large number of questions in each experimental setting subset. There were at least 30 questions sampled for the 13 main experimental settings (i.e., the 3 *context lengths*, the 5 *context domains*, and the 5 *shot-settings*). Due to sampling overlap across these categories, the annotated set was only 220 questions. The results indicate that `InstructGPT` produced high quality out-

Table 4.4 Summary results from the qualitative assessment of `InstructGPT` controllable generated educational questions. Arrows indicate better scores for strictly directional metrics.

Metric	Mean	Standard Deviation
Relevance (↑)	1.0000	0.0000
Grammar (↑)	0.9750	0.1528
Adherence (↑)	0.7341	0.4415
Answerability (↑)	0.8364	0.3708

puts across the varied experimental settings. The questions appear to be nearly always

on topic and grammatically correct (over 97%). For many control elements they are able to reliably stay within the question types they are asked to generate. Additionally, the generated questions were answerable 83% of the time from the input context itself.

In order to demonstrate the power of `InstructGPT`'s education question generation, let's look at a few examples. A paragraph of text from the Wikipedia page on speech recognition was fed to `InstructGPT`. Of the candidates generated from this passage, there are a few great examples of `InstructGPT`'s success. Observe the following examples:

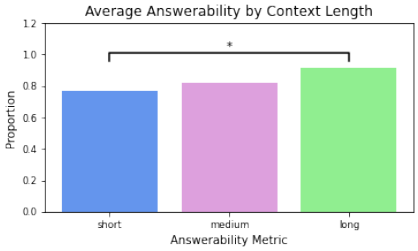
- Remembering: *What is the process of using a person's voice to improve speech recognition accuracy called?*
- Evaluating: *What are the benefits of using speech recognition technology?*
- Advanced: *What are the differences between speaker-dependent and speaker-independent speech recognition systems?*

These examples showcase that `InstructGPT` is able to generate questions spanning multiple parts of the input context and fitting into the question types specified by the *control elements*. The model's varying success across the different experimental parameters included in the qualitative assessment is explored in the subsequent sections. Additional examples of `InstructGPT`'s automatically generated educational questions can be found in Appendix B.4.

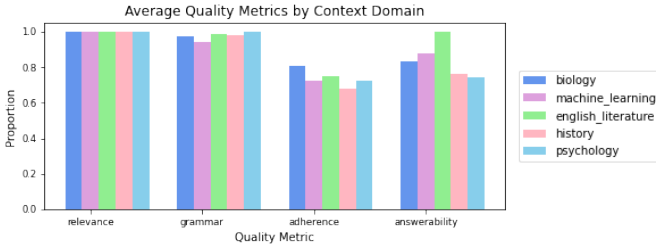
4.2.2 Parameter: Context Length

The *context length* parameter in this experiment represents the length, in number of sentences, of the input passage from which `InstructGPT` generated educational questions. Three possible *context lengths* were included: *short* (1 to 2 sentences), *medium* (3 to 5 sentences), and *long* context passages (6 to 9 sentences). All three length options perform comparably well with respect to *grammar*, *relevance*, and *adherence*.

The *long* contexts stand out because they exhibit higher *answerability*. In fact, there is a statistically significant difference between the *short* and *long* context lengths for *answerability* using Student's t-test and $\alpha = 0.05$. This is an intuitive result as the more material `InstructGPT` has to 'work with', the more likely it's generated questions will be diverse and answerable.



(a) *Answerability* of generated questions from `InstructGPT` by *context length*.



(b) All quality metric results on the generated questions from `InstructGPT` split by *context domain*.

Figure 4.2 Visualizations of the *context length* and *context domain*. Significant difference using Student's t-test and $\alpha = 0.05$ is marked by an asterisk.

4.2.3 Parameter: Context Domain

In order for a question generation system to be robust, it must not be domain-specific. As such, different fields of study, or *context domains*, were included to assess if the results would be skewed if `InstructGPT` excelled or struggled to generate candidates in any particular domain. The preliminary experimentation included five domains: *biology*, *English literature*, *history*, *machine learning*, and *psychology*. The topics for each of the 45 input contexts were selected evenly from the domain set, resulting in 9 contexts each.

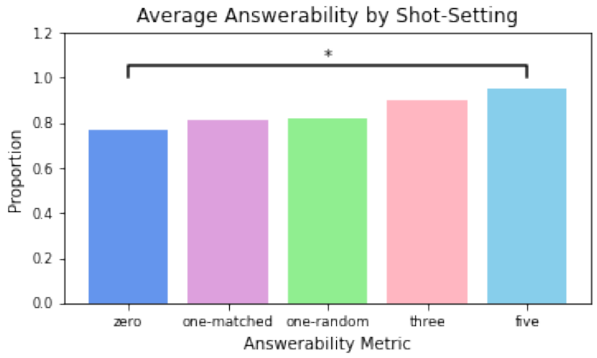
The annotated metrics show some variability in performance depending on the context domain. However, no one domain is significantly or consistently worse across all metrics than any other, as seen in Figure 4.2b. This lack of concrete difference is the desired result, as it implies that this approach works generally for education, and not specifically for a single domain.

4.2.4 Parameter: Shot Setting

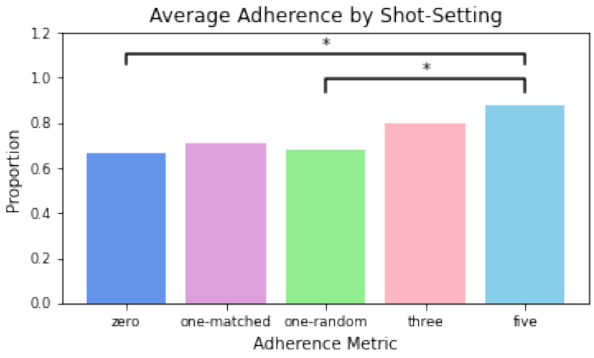
The *shot setting* parameter refers to the idea of few-shot learning introduced in Section 2.3.6. Essentially, few-shot learning includes examples of the desired input output structure directly within a prompt to a LLM. This experiment attempts to perform CTG to generate questions of specific types, that are explained further in Section 4.2.5. Corresponding exemplary question-passage pairs used for few-shot learning were handcrafted to fit into each of these particular question types. In total, 115 example questions were constructed so that there is a question of each type in each domain included in the experiment. Examples of these question-passage pairs can be found in Appendix B.3.

The preliminary experiment included five different *shot settings*: *zero-shot*, *random-one-shot*, *matched-one-shot*, *three-shot* and *five-shot*. The *zero-shot* setting simply queries `InstructGPT` with no examples included in the prompt, only the *control element* and the input context. The one-shot settings include a single example in the query, either one that is in the same domain as the input context, i.e., *matched-one-shot*, or in a random selection from the other four domains, i.e., *random-one-shot*. The *three-shot* and *five-shot* settings' queries include three and five examples respectively, only one of which is in the same domain as the input context.

With respect to *relevance* and *grammar*, there are no distinct differences seen in the various *shot settings*. The differences are found when considering the *answerability* and *adherence* metrics. Here, the few-shot settings outperformed the *zero-shot* setting, with *five-shot* learning performing best, which is aligned with related literature (Liu et al., 2023; Wang et al., 2022b). This is also an intuitive result, since the *five-shot* setting gives more examples of the desired question type for `InstructGPT` to use as additional context in it's prediction of the most probable response. The preliminary experiments only went up to *five-shot* learning, due to resource constraints. Further research could look into higher shot settings.



(a) *Answerability* of generated questions from `InstructGPT` by *shot setting*.



(b) *Adherence* to the *control element* of generated questions from `InstructGPT` by *shot setting*.

Figure 4.3 Visualizations of the *shot setting* key results. Significant differences using Student's t-test and $\alpha = 0.05$ are marked by an asterisk.

4.2.5 Parameter: Control Elements

As explained in Section 2.3.6, one approach to CTG is to use keywords as a *control element*. These keywords work to guide the text generation towards a certain style, sentiment, format, or other. In the case of this preliminary question generation experiment, keyword *control elements* are used to guide the QG towards a certain type of question. These question types are pulled from various question taxonomies from pedagogical literature, introduced in Section 2.1.3. Appendix B.1 includes definitions of all of the question types attempted and their related taxonomies.

The results show that *adherence* to question type with CTG is not perfect - but it is possible. On average, 73.41% of questions in the qualitative assessment adhered to their question type. There was not a large enough sample size to definitively say which question taxonomies are superior to use as control elements for CTG. Of the extensive set of taxonomies tried, two representative taxonomies were chosen for the experiments in Section 4.3: Bloom's taxonomy (Krathwohl, 2002) (which includes *remembering*, *understanding*, *applying*, *analyzing*, *evaluating*, and *creating* question types)

and a difficulty-level taxonomy (which includes *beginner*, *intermediate*, and *advanced* question types) (Pérez et al., 2012).[2] These taxonomies approach the organization of questions in different ways, by the learning goal and by complexity respectively. This creates an interesting comparison among the taxonomic categories to help explore the limits of the CTG approach.

4.3 Teacher's Opinions on the Usefulness of QG with LLMs

The robust QG system glimpsed in Section 4.2 has the potential to empower teachers by decreasing their cognitive load while creating high quality teaching material. It could allow them to easily generate personalized content to fill the needs of different students; for example by adapting questions to Bloom's taxonomy levels (i.e., learning goals) or difficulty levels. These improvements hinge on the assumption that the candidates are high quality and are actually judged to be useful by teachers generally. There is little or no prior work showing a systematic, thorough evaluation of LLMs' ability to generate educational content performed by real-world teachers who will actually be using the generations.[3] This section outlines an experiment with the goal of investigating if LLMs can generate different types of questions from a given context that real teachers think are appropriate for use in the classroom. Our experiment demonstrates that this is the case, with high quality and usefulness ratings across two domains and 9 question types. This evidence of the high quality and usefulness of automatically generated pedagogical questions will hopefully lead to more widespread adoption of CTG in the educational domain, as it is currently an underused resource (Wang et al., 2022a).

[2] See section 2.1.3 for additional details of these taxonomies.

[3] Wang et al. (2022b) show that subject matter experts cannot distinguish between machine written and human written questions, but they state that a future direction is to similarly assess CTG with teachers and students.

4.3.1 Generation with `InstructGPT`

The teacher assessment experiment was conducted with educational question candidates generated in the machine learning (ML) and biology (BIO) domains. There are 68 *long* context passages (6 to 9 sentences) pulled from Wikipedia. From this set, 31 contexts are about machine learning, and 37 are about biology. Using the same handcrafted examples for *five-shot* learning from Section 4.2.4, `InstructGPT` was prompted to generate 612 candidate questions.[4] Each passage has 9 candidates, one with each taxonomic category as the control element.

Overlap of Generated Candidates

There are overlaps within the generated pedagogical questions for this experiment. Specifically, despite having different *control elements*, sometimes the LLM generates the same question for a given context passage twice. Out of 612 candidates, there are 540 unique ones (88.24% are unique). This overlap is low enough that the questions still have the potential to be sufficiently diverse for a teacher's needs. It is important to keep in mind that this overlap is not reflected in the following results, as teachers were asked to rank every candidate independently.

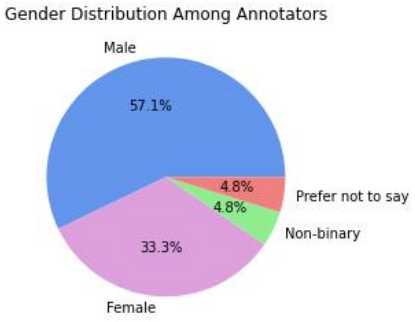
4.3.2 Methodology for Usefulness Study

Annotators There are two cohorts of annotators, one in each domain. The 11 biology annotators have biology teaching experience at least at a high school level. They were recruited on the freelance platform Up Work. The 10 machine learning annotators have computer science, machine learning, artificial intelligence, mathematics or statistics teaching experience at a university level. They were recruited through word of mouth

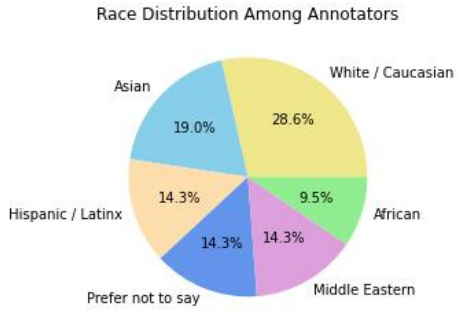
[4]Examples of the passages, few-shot examples, generations, and the question type definitions can be found in Appendix B. Additionally, further details are available here: `https://github.com/sabina-elkins/educational_CQG`

at McGill University and Mila. The assessment experiment is identical for both cohorts. As such, the rest of the experiment is explained in a domain-agnostic manner. The results will be presented separately, as the goal of this work is not to show identical trends between the two domains, but that CTG is appropriate for education in general, not a specific domain.

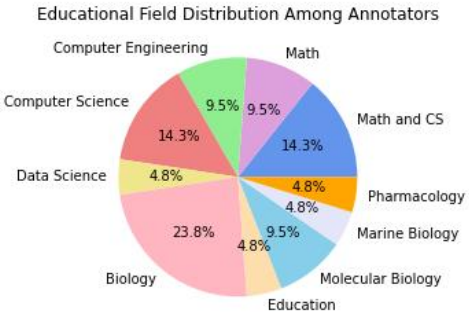
Figure 4.4 shows the diverse demographics of annotators by gender, race, and education (both their field of study and the highest level they attained/are currently en-



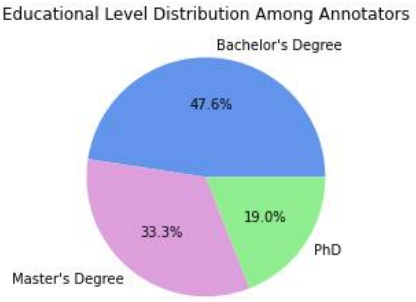
(a) Distribution of participant's gender.



(b) Distribution of participant's race/ethnicity.



(c) Distribution of participant's main field of study.



(d) Distribution of participant's highest educational level attained or in progress.

Figure 4.4 Demographics and educational experience of the participants.

rolled at). These distributions are included to offer transparency about the composition of the participant set, while protecting their personally identifiable data. While the pool of annotators is relatively small, their demographic distribution varies enough to dispel the potential bias in using a set of annotators without diverse backgrounds. All of the annotators are proficient English speakers. The annotator's teaching experience ranges from 1-on-1 tutoring, to hosting lectures for a university course, to being a high school teacher. This diversity of experience enables a more robust assessment of the generated candidates, as teachers with different experience can highlight different valuable aspects of questions that make them *useful*.

Metrics While the main goal of this experiment is to assess the *usefulness* of the generated candidate questions, it is important to also assess their quality. In light of this, the annotation scheme includes the four quality metrics defined in the preliminary assessment: *relevance*, *grammar*, *adherence*, and *answerability*. Their definitions can be found in Section 4.2.1. As the preliminary experiments in Section 4.2 already point to the high quality of the CTG candidates, it is not critical to have multiple annotations of each candidate's quality on all of the metrics. Therefore each annotator was trained to assess the generated candidates on two of the four quality metrics (as well as a *usefulness* metric, explained below). This division reduces both the cognitive load on an individual annotator, and the potential for bias introduced by their confusion between metrics.

The *usefulness* metric is defined by a teacher's answer to the question: "Assume you wanted to teach about context X. Do you think candidate Y would be useful in a lesson, homework, quiz, etc.?" where X is replaced by the context passage, and Y is replaced by the candidate question. This is an ordinal metric with the following four categories:

- *Not useful* (1): The core content of the question is not useful to teach context X at all. For example, the candidate might be off topic, have logical issues, simply not

a useful question to *teach* context X, or be otherwise unacceptable.

- *Useful with major edits* (2): The core content of the question is useful, but the phrasing or presentation of the candidate is not, and would require changes that take more than a minute. For example, the candidate might present an interesting idea that would be useful to teach context X, but the sentence structure is confusing and would need to be completely re-written.
- *Useful with minor edits* (3): The core content of the question is useful, but the phrasing or presentation of the candidate has some minor issues (e.g. grammatical errors, word choice problems) that could be fixed in less than a minute.
- *Useful with no edits* (4): The question is useful as is, and can be used directly without making any changes.

An ideal mean value of 4 means that all of the generated questions are rated as *useful with no edits* by the teachers. Note that the question does not necessarily need to be answerable from the context or adhere to the question type in order to be considered useful. If a teacher rates a question as *not useful* or *useful with major edits* we also ask them to select from a list of reasons why (or write their own).

Pilot Studies Before running the complete experiment, two small pilot studies were conducted. The first pilot was conducted with four annotators in the machine learning domain.[5] The results showed that the annotator training, experiment instructions, and quality metric definitions were clear and agreed upon. However, the first version of the *usefulness* metric introduced ambiguity. This is because the initial idea for the *usefulness* metric was framed as two Likert-scales where the annotator would rate a candidate on (a) if they thought it would be *useful*, and (b) if they thought it would be *useful with minor edit*s. These metrics were overlapping and created some confusion amongst

[5]The annotators in the pilot studies did not also participate in the final experiment.

the annotators. Accordingly, they were combined into the concise ordinal *usefulness* metric explained in Section 4.3.2. A second, smaller pilot with three of the four pilot annotators indicated that this new *usefulness* metric was less ambiguous and as a result more informative.

Reducing Bias Beyond the pilots, there were a few other steps taken to limit the potential for bias in this experiment:

- The order of candidates presented was randomized and only annotators were asked to rate one metric at a time to avoid the Halo effect. The Halo effect is the idea that if an annotator thinks a candidate is high quality in one respect, they then conflate it and rate it highly on the other metrics (Neugaard, 2023). Randomizing the annotation order and separating the metrics is a step towards separating the metric-candidate pairs in the eyes of an annotator, and hopefully reducing the possibility of this conflation.
- Unmarked attention-checking candidates were included in the set of candidates for annotators to rate. These *distractor* questions were obviously wrong (e.g., a random question from a different context, a candidate with injected grammatical errors). Each annotator encountered 12 such questions, to help ascertain if they were paying attention. Any annotators who did not agree on a minimum of 80% of these *distractor* questions were excluded. The annotator's performance on these is further discussed in Section 4.3.3.

4.3.3 Results of Usefulness Study

Annotator Agreement All of the participants annotated candidates from 6 context passages. In order to assess their agreement on the task, they annotated a 7^{th} passage that was the same for all annotators in a given domain cohort. The results for each

metric are reported in Table 4.5. In both domains, *relevance*, *grammar*, and *answerability* have between 85% and 100% observed agreement. The *adherence* metric has lower agreement, between 60% and 80%. Since this metric is more complex than the others and captures the annotators' interpretations of the question taxonomies, this moderate agreement is acceptable and expected.

Unlike the binary metrics, all candidates were rated on *usefulness* by two annotators. As before, only one context passage, the agreement on which is presented in Table 4.5, was seen by all annotators. In both cohorts, the observed agreement on *usefulness* is around 63%. This metric is defined according to a teacher's opinion, and as such is subjective. Thus, the lower agreement between annotators is to be expected. Using Cohen's κ to measure the agreement yields a $\kappa = 0.622$ for the ML cohort and a $\kappa = 0.611$ for the BIO cohort, which implies substantial and moderate agreement respectively (Landis and Koch, 1977). Additionally, the agreement of the annotators on the included *distractor* candidates for this metric (see Section 4.3.2) is $\kappa = 1$ (i.e., perfect agreement), which shows that the annotators agree on the fundamental task but might find different questions useful for their particular approach to teaching.

Quality Metric Results Three quality metrics, *relevance*, *grammar*, and *answerability*, are consistently high for all generated candidates (see in Table 4.5). The fourth quality metric, *adherence*, varies across the taxonomic categories as seen in Figure 4.5a. This variation is similar within the two domains. As might be expected, the categories with more objective definitions are easier for the LLM to generate. For instance, looking only at the 'remembering' category has an *adherence* of 83.3% for the machine learning cohort and 91.7% for the biology cohort. This category is intended to ask for a student to recall a fact or definition. This is might be simple for the LLM to replicate by identifying a relevant text span, and reflects the traditional QG task. By contrast, asking a LLM to generate a 'creating' question is a more open-ended problem, where a text span

Table 4.5 The quality metrics' mean (μ), standard deviation (σ), and observed agreement (i.e., % of the time the annotators chose the same label). The n for the *usefulness* metrics are twice as large because all annotators rated them, unlike with the quality metrics. Arrows indicate better scores for strictly directional metrics.

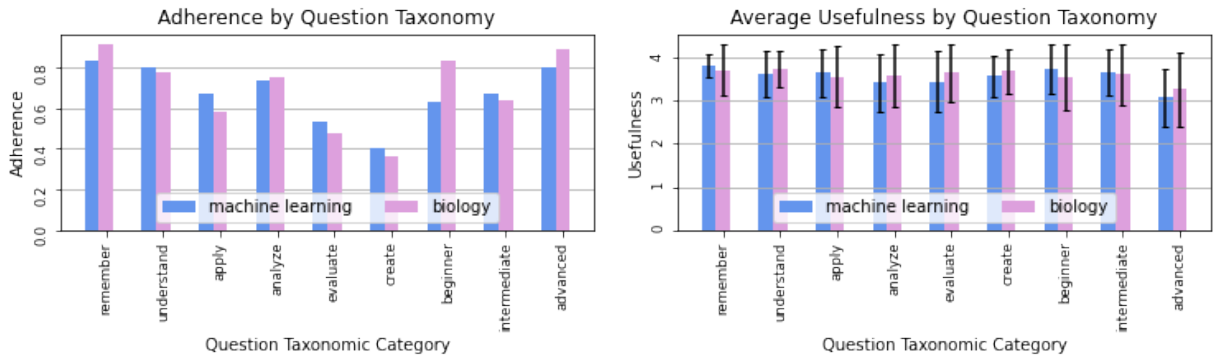
Metric	n	$\mu \pm \sigma$ (ML)	Agreement % (ML)	n	$\mu \pm \sigma$ (BIO)	Agreement % (BIO)
Relevance (↑)	270	0.967±0.180	100	324	0.972±0.165	100
Grammar (↑)	270	0.922±0.268	94.1	324	0.970±0.170	100
Adherence (↑)	270	0.674±0.470	62.2	324	0.691±0.463	79.9
Answerability (↑)	270	0.919±0.274	89.6	324	0.930±0.256	86.7
Usefulness (↑)	539	3.557±0.597	62.7	648	3.593±0.682	62.8

from the context may not be the answer. Accordingly, the model struggles on this less constrained task, and has an *adherence* of only 40.0% for the machine learning cohort and 36.1% for the biology cohort.

Usefulness Metric Results The *usefulness* metric's ordinal categories are mapped from 1 (*not useful*) to 4 (*useful with no edits*). The average usefulness for all candidates is 3.557 for the machine learning cohort and 3.593 for the biology cohort. Note that these averages include two usefulness ratings for each candidate question, which can help to account for individual teacher differences in what they deem *useful*. These differences exist, but are minimal, as per the annotator agreement explained above. Both cohorts having an average *usefulness* of 3.6 is a highly promising result showing that on average teachers find that these generated candidates will be useful in a classroom setting.

There is no significant difference between the usefulness scores of any of the question taxonomy categories, though some variation is present (see Figure 4.5b). On average, each of the question taxonomies are rated between *useful with minor edits* and *useful*

with no edits (i.e., $[3, 4]$). Considering that *adherence* differences across question types, it is important to note that a question which does not adhere to its question taxonomy can still be useful in a different way than intended.



(a) *Adherence* by taxonomic category. (b) Average *usefulness* by taxonomic category.

Figure 4.5 Visualizations of the *usefulness* and *adherence* metrics.

Another key result to note is that 53.4% of the time the reason cited for *not useful* candidates is related to their grammar or phrasing. This can possibly be reduced by a filter that removes malformed questions, but it will lower the available diversity of questions. Alternatively, one can consider CTG as a tool for teachers to use, rather than an independent system needing to create perfect candidates. Approaching CTG in this way leaves room for teachers to choose what kinds of questions to use, and gives them the opportunity to make minor edits to the generated content as they see fit.

Figure 4.5b shows that 'advanced' questions have the lowest *usefulness* scores in both cohorts. Many annotator's reasons for giving a low rating to an 'advanced' question was along the lines of "the question is trying to ask two things at once" or "it should be split into two questions". This follows from the fact that the few-shot learning advanced examples were designed to ask two-part questions to increase the difficulty of the question. For example, the advanced biology question is:

*Can you name **and** describe two useful characteristics of fungi?*

Evidently, some teachers disagree that such a two-part question is useful. Thus, the lower *usefulness* scores for 'advanced' questions might be attributable to the definition of the question type, as opposed to the generation process itself.

4.4 Conclusion

In this chapter, the automatic generation of pedagogical questions was considered. To start, approaches to further the work seen in Chapter 3 by automatically generating question variants were explored. Through some small-scale human evaluations, these candidates were shown to be relatively high quality. Future work would need to replicate the work in Chapter 3 with these automatically generated candidates in order to be more certain of their merit.

Generating variants of questions has limited scope due to the need for existing pedagogical questions. Accordingly, this chapter went on to explore the generation of a variety of questions from source material. This was done using LLMs. Specifically, controllable text generation with `InstructGPT` was conducted to generate questions in different levels of a selection of educational question taxonomies from the same input passage. Then, an annotation experiment designed to explore whether teachers judge these generations as sufficiently high quality and useful for their own classrooms was conducted. The results showed that teachers on average rate these questions as 3.6 out of 4 on the *usefulness* metric; or, in other words, that the teachers see value in these question-type directed generations.

Chapter 5

Conclusion

Educational question generation is a key application of AI in education that has obvious use cases. Previous research has already documented the success of generating pedagogical questions (Wang et al., 2022b), and teacher's acceptance of them (Laban et al., 2022). There is a gap in the research concerning the personalization of QG for education (Kurdi et al., 2020), which this thesis attempts to fill. The two key steps forward are demonstrating the increasing in learning gains students receive when they are given personalized content, and demonstrating a teacher-approved method of personalized exercise generation. Chapter 3 outlines an A/B test which demonstrates the statistically significant increase in solution acceptance, and statistically significant decrease in failure rate that students experience with personalized exercises on an ITS. The results indicate that fitting a question variant to a student's level will improve their success, where a mismatch can be detrimental to student's learning. Chapter 4 outlines the preliminary experimentation which led to a strategy for controllable generation of educational questions, as well as an annotation experiment demonstrating the positive opinions of teachers about these generations. The average *usefulness* is around 3.6 out of 4 for the candidates used in the study, strongly suggesting such generations would be helpful in a classroom setting. These two results work together to

answer the key research question of the thesis: "What is an optimal way to apply and adapt question generation for the educational domain that can improve both the learning and teaching experience?". The results show the benefit personalized pedagogical questions can provide to students, an automatic way to generate such questions, and teacher's opinions that the candidates are highly useful to them.

5.1 Limitations

It is important to acknowledge that all of the automatic content generation conducted in this thesis is limited to the English language. This is due to the fact that related research also focuses generally on English models, and that McGill University and the key researcher are primarily English-speaking. Future research adapting such tools to different languages and mediums (e.g., code rather than natural text) is needed to strengthen the applicability, inclusivity, and power of educational content generation tools.

Additional limitations of this work should be mentioned for both experiments conducted. In Chapter 3, the experiment is limited by not knowing more about the students in two respects: their demographics, and their opinions. As mentioned in Section 3.3, the Korbit platform does not require it's users to disclose any information about themselves other than a name and email address, so the demographics of the participant pool are unknown. It is possible that this is a source of bias. As per the student's opinions, this experiment would benefit from a survey to compare their thoughts on the learning experience, as opposed to solely relying on their performance. Knowing about student preference would strengthen the argument for personalization within an ITS. In Chapter 4, the experiment takes steps towards demonstrating the realistic usefulness of applying CTG to generate educational questions, but it is not a complete proof. The limitations of this work include the single LLM considered and the inde-

pendence assumption explained in Section 4.3.1. Additionally, it will be important to expand the usefulness judgements to include the other group of people involved in the learning process: students. Applying these generated candidates to form real-world lessons and evaluate their impact on learning outcomes will demonstrate their ultimate value.

5.2 Future Directions

The steps taken in Chapter 4 to prove the usefulness of automatically generated educational questions are key to their widespread adoption in real classroom settings. But the work does not stop there. A follow-up work by the author (currently in progress) takes the work in Section 4.3 a step further by removing the independence assumptions. Instead of simply assessing if the questions are *useful* in isolation, this new work assesses their ability to be used in a quiz writing process. Such proof of applicability will further create research-backing for the application of QG to the educational domain giving confidence to teachers and policy makers alike in the power of these tools.

Additional future directions exist for this line of work with respect to the other key player in the use of automatically generated pedagogical questions: students. The work in Chapter 4 demonstrates that teachers value the generated candidates, but fails to speak to how they might affect students. Future experiments should explore how these machine-written questions are interpreted by students, either by assessing if they can even tell them apart from their human-written counterparts (as seen with domain experts in Wang et al. (2022b)) or how student performance is affected by their use (as a follow up to the work in Chapter 3). An experiment of this type will likely be resource expensive, but also critical to completing the understanding of how automatically generated educational questions affect the real classroom setting.

Appendix A

Variant Generation Examples

Table A.1: Examples of exercise variants simplified with ACCESS.

Success?	Original Question	Generated Variant	Explanation
✓	What is the purpose of converting to a standard normal distribution?	What is the purpose of changing to a standard normal distribution?	Word change to a simpler version.
✓	If we see there is a lot of scatter in the values of a numeric variable (e.g. with a histogram or a boxplot), what can we say about this variable's variance?	If we see there is a lot of scatter in the values of a numeric variable, what can we say about this variable's variance?	Removal of unnecessary information.
✓	Can you name one metric used to evaluate the linear regression model?	Can you name a metric used to study linear regression?	Word change(s) to a simpler version.
X	What type of graph would you use to show the relationship between the average income for adults and the number of years of education completed?	What type of graph would you use to show the relationship between the average income for adults and the number of years of education completed?	The model failed to make any simplification.
X	Suppose you have two investment options: Option A has an expected profit of 500,000 USD and Option B has an expected profit of 450,000 USD. If both options have the same variance, which investment option should you choose?	Suppose you have two types of money: Option A has a profit of 500,000 USD and Option B has an expected profit of 450,000 USD. If both options have the same variance, you should give you the option to choose?	The attempted simplification made the last sentence nonsensical.

Table A.2: Examples of elaboration generation exercise variants.

Success?	Original Question	Generated Variant	Explanation
✓	What is the purpose of converting to a standard normal distribution?	What is the purpose of converting to a standard normal distribution? **Standard normal distribution** = A normal distribution with a mean of 0 and a standard deviation of 1, written $Z \sim N(0, 1)$.	Addition of definition provides additional context to the student.
✓	You are given a dataset with monthly sales and advertising costs in dollars. If you plot the data on a scatter plot, how will you label the x and y axes?	You are given a dataset with monthly sales and advertising costs in dollars. If you plot the data on a scatter plot, how will you label the x and y axes? **Scatter plot** = A graph that uses dots to represent values for 2 numeric variables. It is used to observe relationships between the numeric variables.	Addition of definition provides additional context to the student.
X	Consider the following data set: $\{47, 43, 41, 39, 41, 36, 43\}$. Find the third ($Q_3$) quartile.	Consider the following data set: $\{47, 43, 41, 39, 41, 36, 43\}$. Find the third ($Q_3$) quartile. **Set** = A well-defined collection of distinct objects. In set theory, a set is often denoted by a capital letter.	The definition adds little to no value to a student in this case.
X	What does the threshold value represent?	What does the threshold value represent? **Threshold** = A value beyond which there is a change in the manner an algorithm proceeds.	The definition trivializes the question in this case.

Table A.3: Examples of paraphrased exercises using `Parrot`.

Success?	Original Question	Generated Variant	Explanation
✓	Explain the motivation of prioritizing aging receivables in the context of evaluating late customers.	Why are aging receivables prioritized in the context of evaluating late customers?	Rephrasing a statement into a question.
✓	What approach can be used to get the first 5 rows of any Series where it exists?	What is the easiest way to get the first 5 rows of a series?	Simplification of the wording.
✓	What does a machine learning algorithm **learn** from in general?	What can machine learning algorithms learn from?	Simplification of the wording. Adds more abstraction (by removing 'in general') allowing the student to respond with more than one answer.

X	Consider the following data set: $\{47, 43, 41, 39, 41, 36, 43\}$. Find the third ($Q_3$) quartile.	Consider the following data set: $\{47, 43, 41, 39, 41, 36, 43\}$. Find the third quartile of the q3.	Injection of grammatical errors makes the final sentence unreadable.
X	What is the significance of there being 100 unique customer IDs and 100 repeat customers?	What is the significance of 100 unique customer IDs?	The variant is still grammatically correct but has lost the meaning of the original question.

Appendix B

Generating Educational Questions with `InstructGPT`

B.1 Question Types and Taxonomies

This table includes all of the question types used in the preliminary experiments in Section 4.2. Only the Bloom's Taxonomy and some of the Level Taxonomy prompts are used in the experiments in Section 4.3.

Table B.1: Question types used in the preliminary experiment in Section 4.2

Question Type	Taxonomy	Definition
remembering	Bloom's	The question should ask students to retrieve from memory a fact, term, concept, etc..
recall	Bloom's	Sub-categories of the remembering category of Bloom's Taxonomy.
recognition	Bloom's	Sub-categories of the remembering category of Bloom's Taxonomy.
understanding	Bloom's	The question should ask students to demonstrate their understanding of material by describing, explaining, comparing, interpreting, etc.
comparison	Bloom's	Sub-categories of the understanding category of Bloom's Taxonomy.
definition	Bloom's	Sub-categories of the understanding category of Bloom's Taxonomy.
example	Bloom's	Sub-categories of the understanding category of Bloom's Taxonomy.
applying	Bloom's	The question should ask students to use the presented concepts to solve problems, or explain ideas in a different way.
analyzing	Bloom's	The question should ask students to break material into parts, and/or show how different ideas relate to one another.

evaluating	Bloom's	The question should ask students to give opinions on, make judgments about, or interpret meaning from material.
creating	Bloom's	The question should ask students to combine material together in a different way than it was presented.
beginner	Level	The question should be posed so that the correct answer is a simple span from the input context (often a single concept or a list).
easy	Level	Alternate keyword for the beginner question type.
intermediate	Level	The question should be posed so that the correct answer is a span from the input context that is more complex than a single concept (eg. an explanation or an example), or requires understanding on the part of the student to arrive at a simple answer.
advanced	Level	The question should be posed so that the correct answer requires a student's rephrasing of multiple parts of the input, or the answer must require independent thought.
hard	Level	Alternate keyword for the advanced question type.
true or false	Response	The question should be posed so that the correct answer must be either 'true' or 'false'.
yes or no	Response	The question should be posed so that the correct answer must be either 'yes' or 'no'.
explanation	Response	The question should be posed so that the correct answer must be an explanation of a concept or fact.
opinion	Response	The question should be posed so that the correct answer must be an opinion.
what	Question word	The question should be posed so that it includes the 'what' question word.
why	Question word	The question should be posed so that it includes the 'why' question word.
how	Question word	The question should be posed so that it includes the 'how' question word.

B.2 Contexts

All of the contexts from used in the human evaluation in Section 4.3 for generation can be found here: `https://github.com/sabina-elkins/educational_CQG` (originally from Wikipedia (Wikipedia contributors)). The following is only a subset of the contexts to demonstrate the general idea.

Table B.2: Examples of contexts used for educational question generation.

Domain	Length	Context
Biology	Short	In molecular biology, the term double helix refers to the structure formed by double-stranded molecules of nucleic acids such as DNA. The double helical structure of a nucleic acid complex arises as a consequence of its secondary structure, and is a fundamental component in determining its tertiary structure.
English Literature	Medium	In literature, Romanticism found recurrent themes in the evocation or criticism of the past, the cult of 'sensibility' with its emphasis on women and children, the isolation of the artist or narrator, and respect for nature. Furthermore, several romantic authors, such as Edgar Allan Poe, Charles Maturin and Nathaniel Hawthorne, based their writings on the supernatural/occult and human psychology. Romanticism tended to regard satire as something unworthy of serious attention, a prejudice still influential today. The Romantic movement in literature was preceded by the Enlightenment and succeeded by Realism.
Machine Learning	Long	Decision tree learning uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). It is one of the predictive modelling approaches used in statistics, data mining and machine learning. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data (but the resulting classification tree can be an input for decision making). This page deals with decision trees in data mining.

B.3 Few-Shot Examples

Note that all of the few-shot examples used in the human evaluation in Section 4.3 can be seen here: `https://github.com/sabina-elkins/educational_CQG`. The following is only a subset to demonstrate the general idea. All of these few-shot examples were constructed from the following context from Wikipedia (Wikipedia contributors) for the ML domain:

Artificial neural networks (ANNs), usually simply called neural networks (NNs) or neural

nets, are computing systems inspired by the biological neural networks that constitute animal brains. An ANN is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal to other neurons. An artificial neuron receives signals then processes them and can signal neurons connected to it. The 'signal' at a connection is a real number, and the output of each neuron is computed by some non-linear function of the sum of its inputs.

Table B.3: Examples of hand-crated questions used for few-shot learning within the educational question generation prompts.

Question Type	**Hand-Crafted Example**
Remembering (Bloom's Taxonomy)	What is another name for an artificial neural network?
Understanding (Bloom's Taxonomy)	How are 'signals' represented in an artificial neural network?
Applying (Bloom's Taxonomy)	What would you do to send a signal through an artificial neuron?
Analyzing (Bloom's Taxonomy)	Can you compare the way a signal travels through a brain and through an artificial neural network?
Evaluating (Bloom's Taxonomy)	Do you think artificial neural networks are good models of the human brain?
Creating (Bloom's Taxonomy)	What metaphor, other than the brain, can you use to describe an artificial neural network?
Beginner	What do you call computing systems that are inspired by the animal brain?
Intermediate	What are artificial neurons within a neural network?
Advanced	What are the inputs and outputs of an artificial neuron?

B.4 Examples of Generations

For reproducibility's sake, all of the generated candidates used in the human evaluation in Section 4.3 are here: `https://github.com/sabina-elkins/educational_CQG`. The following table includes a few selected examples of where the model succeeds and fails. Note that there are question types below which are included in the preliminary experiments in Section 4.2 but not in 4.3.

Table B.4: Examples of automatically generated educational questions using few-shot learning and `InstructGPT`.

Domain	Question Type	Context	Candidate
Machine Learning	True or False	Decision tree learning uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). It is one of the predictive modelling approaches used in statistics, data mining and machine learning. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data (but the resulting classification tree can be an input for decision making). This page deals with decision trees in data mining.	True or False: Decision tree learning is used to represent decisions and decision making in data mining.
Biology	Comparison	Sex linked describes the sex-specific patterns of inheritance and presentation when a gene mutation (allele) is present on a sex chromosome (allosome) rather than a non-sex chromosome (autosome). In humans, these are termed X-linked recessive, X-linked dominant and Y-linked. The inheritance and presentation of all three differ depending on the sex of both the parent and the child. This makes them characteristically different from autosomal dominance and recessiveness. There are many more X-linked conditions than Y-linked conditions, since humans have several times as many genes on the X chromosome than the Y chromosome. Only females are able to be carriers for X-linked conditions; males will always be affected by any X-linked condition, since they have no second X chromosome with a healthy copy of the gene. As such, X-linked recessive conditions affect males much more commonly than females.	How do X-linked conditions differ from autosomal dominance and recessiveness?

History	Remembering	The Egyptian pyramids are ancient masonry structures located in Egypt. Sources cite at least 118 identified "Egyptian" pyramids. Approximately 80 pyramids were built within the Kingdom of Kush, now located in the modern country of Sudan. Of those located in modern Egypt, most were built as tombs for the country's pharaohs and their consorts during the Old and Middle Kingdom periods. The earliest known Egyptian pyramids are found at Saqqara, northwest of Memphis, although at least one step-pyramid-like structure has been found at Saqqara, dating to the First Dynasty: Mastaba 3808, which has been attributed to the reign of Pharaoh Anedjib, with inscriptions, and other archaeological remains of the period, suggesting there may have been others.	What country is the Kingdom of Kush located in today?
English Literature	What	In literature, Romanticism found recurrent themes in the evocation or criticism of the past, the cult of 'sensibility' with its emphasis on women and children, the isolation of the artist or narrator, and respect for nature. Furthermore, several romantic authors, such as Edgar Allan Poe, Charles Maturin and Nathaniel Hawthorne, based their writings on the supernatural/occult and human psychology. Romanticism tended to regard satire as something unworthy of serious attention, a prejudice still influential today. The Romantic movement in literature was preceded by the Enlightenment and succeeded by Realism.	What themes were commonly explored in Romantic literature?
Psychology	Advanced	Cognitive neuroscience is the scientific field that is concerned with the study of the biological processes and aspects that underlie cognition, with a specific focus on the neural connections in the brain which are involved in mental processes. It addresses the questions of how cognitive activities are affected or controlled by neural circuits in the brain. Cognitive neuroscience is a branch of both neuroscience and psychology, overlapping with disciplines such as behavioral neuroscience, cognitive psychology, physiological psychology and affective neuroscience.	What are the implications of cognitive neuroscience for understanding how neural circuits in the brain influence mental processes?

Bibliography

Manish Agarwal and Prashanth Mannem. 2011. Automatic gap-fill question generation from text books. In *Proceedings of the sixth workshop on innovative use of NLP for building educational applications*, pages 56–64.

Haldun Akoglu. 2018. User's guide to correlation coefficients. *Turkish Journal of Emergency Medicine*, 18(3):91–93.

Ali Alkhatlan and Jugal Kalita. 2018. Intelligent tutoring systems: A comprehensive historical survey with recent developments. *CoRR*, abs/1812.09628.

Evelyn Ashton-Jones. 1988. Asking the right questions: A heuristic for tutors. *The Writing Center Journal*, 9(1):29–36.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

David Baidoo-Anu and Leticia Owusu Ansah. 2023. Education in the era of generative artificial intelligence (ai): Understanding the potential benefits of chatgpt in promoting teaching and learning. *Available at SSRN 4337484*.

Barker Bausell, William Moody, and Neil Walzl. 1972. A factorial study of tutoring versus classroom instruction. *American Educational Research Journal*, 9(4):591–597.

Benjamin S Bloom. 1956. *Taxonomy of educational objectives: The classification of educational goals: By a committee of college and university examiners*. David McKay.

Benjamin S Bloom. 1984. The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational researcher*, 13(6):4–16.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Jack A Chambers and Jerry W Sprecher. 1983. *Computer-assisted instruction: Its use in the classroom*. Prentice Hall Direct.

Lijia Chen, Pingping Chen, and Zhijian Lin. 2020. Artificial intelligence in education: A review. *Ieee Access*, 8:75264–75278.

Maria Chinkina and Detmar Meurers. 2017. Question generation for language learning: From ensuring texts are read to supporting learning. In *Proceedings of the 12th workshop on innovative use of NLP for building educational applications*, pages 334–344.

Prithiviraj Damodaran. 2021. Parrot: Paraphrase generation for nlu.

Bidyut Das, Mukta Majumder, Santanu Phadikar, and Arif Ahmed Sekh. 2021. Automatic question generation and answer assessment: a survey. *Research and Practice in Technology Enhanced Learning*, 16(1):1–15.

Richard R Day and Jeong-suk Park. 2005. Developing reading comprehension questions. *Reading in a foreign language*, 17(1):60–73.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chenhe Dong, Yinghui Li, Haifan Gong, Miaoxin Chen, Junxin Li, Ying Shen, and Min Yang. 2022. A survey of natural language generation. *ACM Computing Surveys*, 55(8):1–38.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. *arXiv preprint arXiv:1705.00106*.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Sian Gooding and Ekaterina Kochmar. 2019. Recursive context-aware lexical simplification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4855–4865, Hong Kong, China. Association for Computational Linguistics.

Arthur C Graesser. 2011. Learning, thinking, and emoting with discourse technologies. *American psychologist*, 66(8):746.

Arthur C Graesser, Patrick Chipman, Brian C Haynes, and Andrew Olney. 2005. Autotutor: An intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions on Education*, 48(4):612–618.

Arthur C Graesser and Natalie K Person. 1994. Question asking during tutoring. *American educational research journal*, 31(1):104–137.

John Hattie. 2012. *Visible learning for teachers: Maximizing impact on learning*. Routledge.

Mariane Hedegaard. 2012. The zone of proximal development as basis for instruction. In *An introduction to Vygotsky*, pages 234–258. Routledge.

Michael Heilman and Noah A Smith. 2010a. Good question! statistical ranking for question generation. In *Human language technologies: The 2010 annual conference of the North American Chapter of the Association for Computational Linguistics*, pages 609–617.

Michael Heilman and Noah A Smith. 2010b. Rating computer-generated questions with mechanical turk. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's mechanical turk*, pages 35–40.

Eric Horvitz. 2022. On the horizon: Interactive and compositional deepfakes. In *Proceedings of the 2022 International Conference on Multimodal Interaction*, pages 653–661.

Stefan Hrastinski, Stefan Stenbom, Simon Benjaminsson, and Malin Jansson. 2021. Identifying and exploring the effects of different types of tutor questions in individual online synchronous tutoring in mathematics. *Interactive Learning Environments*, 29(3):510–522.

Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. 2023. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274.

Randa Amer Khella and Samy S Abu-Naser. 2018. An intelligent tutoring system for teaching french.

Calvin L King, Harco LHS Warnars, Nurulhuda Nordin, and Wiranto H Utomo. 2021. Intelligent tutoring system: learning math for 6th-grade primary school students. *Education Research International*, 2021:1–10.

Ekaterina Kochmar, Dung Do Vu, Robert Belfer, Varun Gupta, Iulian Vlad Serban, and Joelle Pineau. 2020. Automated personalized feedback improves learning gains in an intelligent tutoring system. In *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21*, pages 140–146. Springer.

Korbit. About korbit. *Korbit*.

David R Krathwohl. 2002. A revision of bloom's taxonomy: An overview. *Theory into practice*, 41(4):212–218.

James A Kulik and JD Fletcher. 2016. Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of educational research*, 86(1):42–78.

Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2020. A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, 30:121–204.

Philippe Laban, Chien-Sheng Wu, Lidiya Murakhovs' ka, Wenhao Liu, and Caiming Xiong. 2022. Quiz design task: Helping teachers create quizzes with automated question generation. *arXiv preprint arXiv:2205.01730*.

Susanne P Lajoie and Alan Lesgold. 1989. Apprenticeship training in the workplace: Computer-coached practice environment as a new form of apprenticeship.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Jill H Larkin and Ruth W Chabay. 1992. *Computer assisted instruction and intelligent tutoring systems: Shared goals and complementary approaches*. Routledge.

Thomas J. II Lasley, editor. 2023. *Bloom's taxonomy*, pages 1–1. Encyclopedia Britannica, Inc.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Chao-Yi Lu and Sin-En Lu. 2021. A survey of approaches to automatic question generation: from 2019 to early 2021. In *Proceedings of the 33rd Conference on Computational Linguistics and Speech Processing (ROCLING 2021)*, pages 151–162.

Stephen Marche. 2022. The college essay is dead. *The Atlantic*.

Louis Martin, Benoît Sagot, Eric de la Clergerie, and Antoine Bordes. 2019. Controllable sentence simplification. *arXiv preprint arXiv:1910.02677*.

Roger Pizarro Milian and Rachel Janzen. 2023. How are canadian postsecondary students using chatgpt? *Academica Group*.

B Neugaard. 2023. *Halo Effect*. Encyclopedia Britannica.

OpenAI. 2022. *Introducing ChatGPT*.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Edwin A. Peel. 2023. *Pedagogy*. Encyclopedia Britannica.

Elena Verdú Pérez, Luisa M Regueras Santos, María Jesús Verdú Pérez, Juan Pablo de Castro Fernández, and Ricardo García Martín. 2012. Automatic classification of question difficulty level: Teachers' estimation vs. students' perception. In *2012 Frontiers in Education Conference Proceedings*, pages 1–5. IEEE.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *CoRR*, abs/1802.05365.

Miquel Porta. 2016. *Relative Risk Reduction (RRR)*. Oxford University Press.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Claude Sammut and Geoffrey I. Webb, editors. 2017. *A/B Testing*, pages 1–1. Springer US, Boston, MA.

Iulian Vlad Serban, Varun Gupta, Ekaterina Kochmar, Dung D Vu, Robert Belfer, Joelle Pineau, Aaron Courville, Laurent Charlin, and Yoshua Bengio. 2020. A large-scale, open-domain, mixed-interface dialogue-based its for stem. In *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21*, pages 387–392. Springer.

Megha Srivastava and Noah Goodman. 2021. Question generation for adaptive education. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 692–701, Online. Association for Computational Linguistics.

Francois St-Hilaire, Nathan Burns, Robert Belfer, Muhammad Shayan, Ariella Smofsky, Dung Do Vu, Antoine Frau, Joseph Potochny, Farid Faraji, Vincent Pavero, et al. 2021. A comparative study of learning outcomes for online learning platforms. In *Artificial Intelligence in Education: 22nd International Conference, AIED 2021, Utrecht, The Netherlands, June 14–18, 2021, Proceedings, Part II*, pages 331–337. Springer.

Francois St-Hilaire, Dung Do Vu, Antoine Frau, Nathan Burns, Farid Faraji, Joseph Potochny, Stephane Robert, Arnaud Roussel, Selene Zheng, Taylor Glazier, et al. 2022. A new era: Intelligent tutoring systems will transform online learning for millions. *arXiv preprint arXiv:2203.03724*.

Yih Tyng Tan and Abdul Rahman Othman. 2013. The relationship between complexity (taxonomy) and difficulty. *AIP Conference Proceedings*, 1522(1):596–603.

Robert S Taylor. 1962. The process of asking questions. *American documentation*, 13(4):391–396.

Christian Terwiesch. 2023. Would chat gpt3 get a wharton mba? a prediction based on its performance in the operations management course. *Mack Institute for Innovation Management at the Wharton School, University of Pennsylvania*.

Sai Vamsi, Venkata Balamurali, K Surya Teja, and Praveen Mallela. 2020. Classifying difficulty levels of programming questions on hackerrank. In *Advances in Decision Sciences, Image Processing, Security and Computer Vision: International Conference on Emerging Trends in Engineering (ICETE), Vol. 1*, pages 301–308. Springer.

Rachel Van Campenhout, Martha Hubertz, and Benny G Johnson. 2022. Evaluating ai-generated questions: A mixed-methods analysis using question data and student perceptions. In *Artificial Intelligence in Education: 23rd International Conference, AIED 2022, Durham, UK, July 27–31, 2022, Proceedings, Part I*, pages 344–353. Springer.

Kurt VanLehn. 2011. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational psychologist*, 46(4):197–221.

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.

Xu Wang, Simin Fan, Jessica Houghton, and Lu Wang. 2022a. Towards process-oriented, modular, and versatile question generation that meets educational needs. *arXiv preprint arXiv:2205.00355*.

Zichao Wang, Jakob Valdez, Debshila Basu Mallick, and Richard G Baraniuk. 2022b. Towards human-like educational question generation with large language models. In *Artificial Intelligence in Education: 23rd International Conference, AIED 2022, Durham, UK, July 27–31, 2022, Proceedings, Part I*, pages 153–166. Springer.

Marilyn Domas White and Mirja Iivonen. 2002. Assessing level of difficulty in web search questions. *The Library Quarterly: Information, Community, Policy*, 72(2):205–233.

Joseph B. Wiggins, Kristy Elizabeth Boyer, Alok Baikadi, Aysu Ezen-Can, Joseph F. Grafsgaard, Eun Young Ha, James C. Lester, Christopher M. Mitchell, and Eric N. Wiebe. 2015. Javatutor: An intelligent tutoring system that adapts to cognitive and affective states during computer programming. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, SIGCSE '15, page 599, New York, NY, USA. Association for Computing Machinery.

Wikipedia contributors. Wikipedia, the free encyclopedia.

John H Wolfe. 1976. Automatic question generation from text-an aid to independent study. In *Proceedings of the ACM SIGCSE-SIGCUE technical symposium on Computer science and education*, pages 104–112.

Xuchen Yao and Yi Zhang. 2010. Question generation with minimal recursion semantics. In *Proceedings of QG2010: The Third Workshop on Question Generation*, pages 68–75. Citeseer.

Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. 2022. A survey of controllable text generation using transformer-based pre-trained language models. *arXiv preprint arXiv:2201.05337*.

Ruqing Zhang, Jiafeng Guo, Lu Chen, Yixing Fan, and Xueqi Cheng. 2021. A review on question generation from natural language text. *ACM Transactions on Information Systems (TOIS)*, 40(1):1–43.