TechBot: A Computer Science Focused QA System

Alisha Rauniyar[1], Ankit Shrestha[2], Ayush Tamang[3], Alaka Rai[4], Ajay Mani Paudel[5]

[1]*Department of Computer and Electronics Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal,* alisha.rauniyarals@gmail.com
[2]*Department of Computer and Electronics Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal,* ankitshrestha008@gmail.com
[3]*Department of Computer and Electronics Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal,* ayutmg456@gmail.com
[4]*Department of Computer and Electronics Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal,* mrsalka02@gmail.com
[5] *Supervisor, Department of Computer and Electronics Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal,* ajayamani@gmail.com

Abstract

This paper presents Tech Bot: A Computer Science Focused QA System designed for the computer science domain. Since there is a rapid increase in the field of computer science, a system that can provide accurate, context-aware responses specific to computer science is desperately needed. TechBot addresses this need by combining the advanced capabilities of Natural Language Processing techniques and Deep Learning Models. It includes a strong preprocessing pipeline, a query comprehension module that is capable of decoding technical terms and an answer retrieval framework optimized for computer science content. It performs better in the computer science domain than in general-purpose systems. Using the Hugging Face Dataset Library with extra web scraping, 20,076 question-answer pairs were collected. Secondly, the QA pairs were split into training and validation sets. Two state-of-the-art transformer-based sequence-to-sequence models, T5 and BART, which are designed for natural language generation, were employed. After fine-tuning on the domain-specific dataset, BART outperformed T5 based on the ROUGE score evaluated on the validation set. Initially, the BART model with 140 million parameters was trained for 21 epochs, achieving a ROUGE score of 0.2867. Finally, an average ROUGE score of 0.2890 was obtained on the test dataset.

Keywords: Chatbot, Web Scraping, Transformers, Language transformation, Natural Language Processing, Computer Science

1. Introduction

The rapid advancement of technology and escalating complexity of computer science (CS) concepts in the 21st century have underscored the inadequacy of traditional learning methods, such as books and articles, in addressing the diverse needs of learners and professionals. As the field expands across domains like machine learning, data science, and software engineering, accessing accurate, context-aware guidance has become increasingly challenging, leading to fragmented and inefficient learning processes. To address this gap, TechBot: A Computer Science Focused QA system is introduced that leverages natural language processing (NLP) and machine learning to deliver precise, user-centric explanations of CS concepts. By synthesizing theoretical knowledge and practical applications, TechBot streamlines the extraction of relevant information from expansive knowledge bases, offering interactive, context-sensitive responses that enhance learning efficiency. This system addresses critical limitations in conventional resources—such as outdated content and oversimplification while mitigating the challenges of navigating rapidly evolving technical literature, thereby empowering users to bridge the divide between conceptual understanding and real-world implementation in an increasingly complex discipline.

2. Related work

Recent research in domain-specific question-answering (QA) systems underscores the critical role of integrating specialized knowledge to improve accuracy and contextual relevance. In the medical domain, (A. Ekbal, 2024) introduced KI-MAG, a BART-based abstractive QA system infused with medical knowledge, which achieved a 15% improvement in BLEU scores over conventional models. However, its reliance on static domain knowledge limits performance on emerging or rare medical conditions, highlighting challenges in dynamic knowledge adaptation. Parallel advancements in general-purpose QA include (J. J. Bird, 2024), who demonstrated the effectiveness of T5-based data augmentation and ensemble learning, achieving 99.59% classification accuracy. Their work emphasizes the utility of transformer-based paraphrasing for dataset enrichment but remains confined to classification tasks rather than generative QA. Within computer science, domain-specific systems have predominantly focused on retrieval-based approaches. (Chen, 2022) developed CodeQA, alBERT-driven system for programming tutorials, which excels in answer retrieval but lacks generative capabilities. Similarly, (Smith, 2023) proposed a BERT-based QA framework tailored to software engineering, yet their work prioritizes keyword matching over contextual synthesis. These systems, while effective for structured queries, struggle with abstractive tasks requiring nuanced understanding of complex concepts.

3. Methodology

3.1. Working mechanism

(Figure 1) is represented as a block diagram of TechBot: A Computer Science focused QA System. The development of this project is included in multiple phases, and each phase is ensured to make this model efficient and intelligent enough to simplify computer science related topics. The process is begun with a collection of training dataset that is subjected to various preprocessing steps. The dataset is contained with 20,076 questions along with their respective answers. The question-answer pairs related to computer science topics are trained in models to generate answers to questions.

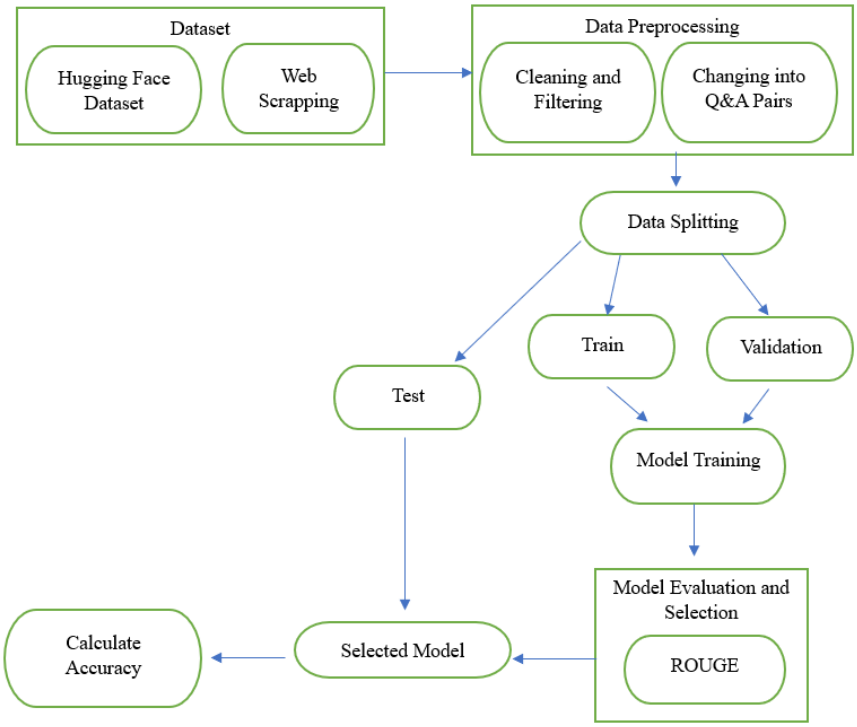

Figure 1: Block diagram of the TechBot

3.1.1 Data Collection

For any project, dataset collection is considered one of the most important phases as accuracy and reliability of the project are helped to be ensured. For this project, the dataset is collected from different sources that include pre-existing datasets and additional web scraping. Pre-existing datasets are extracted from the

Hugging Face dataset library, which is found to provide well-structured question-answer pairs related to various domains like natural language processing, data science, software engineering, artificial intelligence, Python, and machine learning. Altogether, 5,342 question-answer pairs are contained. Since these datasets are not found to be sufficient to enhance the knowledge base of the project, web scraping is further decided to be performed to extract data from educational websites like GeeksforGeeks and W3Schools, where computer science related information is contained. Python's 'request' module is first imported in a bid to acquire HTML content. This is used to make HTTP 'GET' calls on URLs of interest. 'BeautifulSoup' is then utilized for analyzing raw HTML content and major aspects such as a page heading, title, and body content for further processing.

3.1.2 Data Preprocessing

Data preprocessing is regarded as one of the most important steps to ensure the reliability and accuracy of the model so that a desirable result is provided. As the dataset is collected from two different sources, preprocessing steps are required.

The dataset from the Hugging Face dataset library is found in structured form and only basic preprocessing steps are required. It is made available in questions along with their respective answers. The preprocessing steps are included such as checking for null values, removing duplicate entries, and checking for the consistency of the content. After the completion of the cleaning process, all of these datasets are combined into a single CSV file, resulting in 5,342 question and answer pairs.

The dataset obtained from the web scraping process is required to undergo many preprocessing steps as it is obtained in unstructured format. It is consisted of page heading, title, and context rather than direct question and answer pairs. From scraping W3Schools, 10,392 rows are obtained in the CSV file. Similarly, from GeeksForGeeks, 23,384 rows are included. By combining these two CSV files using concatenation, 33,775 rows are obtained. These rows are not exactly the data required for training the model; thus, a significant amount of cleaning and preprocessing is needed. At first, rows with irrelevant values like HTML tags, advertisements, navigation links, and duplicate content are removed using Python's "Pandas" library to ensure only meaningful content is retained. Since raw scraped data is still in the format of page heading, title, and context, further rephrasing is required. Page headings with their titles are converted into questions, and their respective context into answers. Through this method, 5,915 question and answer pairs are collected. In addition, the DeepSeek-R1 reasoning model (a 14-billion parameter model) is employed to rephrase title and context into question-and-answer pairs, and around 8,819 pairs are obtained. Lastly, by merging all these question-and-answer pairs, 20,076 distinct pairs of questions and answers are obtained for model training.

3.1.3 Model Selection and Training

Model selection is referred to as the process of selecting an appropriate machine learning model or deep learning model suitable for a project. To select the best model that is able to fulfill the requirements of the project, it is considered very necessary that different models are evaluated based on architecture, size, and performance. There are lots of factors by which the selection of a model is influenced, such as the size of the dataset, the capacity of the model to handle complexity, task-specific requirements, etc.

Model training is referred to as the phase where learning from the labeled dataset is done by the machine through the adjustment of all internal parameters to provide accurate results by minimizing errors and improving efficiency. For this project, a model that could handle complex computer science-related question answers and could be fine-tuned by the dataset is needed. To build the robust and efficient question-answer model, two transformer-based models — Bidirectional and Auto-Regressive Transformer (BART) and Text-to-Text Transfer Transformer (T5) — are selected.

3.1.3.1 Bidirectional and Auto-Regressive Transformer

BART is described as a transformer model that is developed by Facebook AI and is used by leveraging the strengths of Generative Pre-trained Transformer (GPT) and Bidirectional Encoder Representations from Transformers (BERT). BART is considered a sequence-to-sequence model with an encoder-decoder architecture. The encoder is used to process the input text (BERT), and the decoder is used to generate new

text (GPT). This unique combination of BART is allowed to understand the full context of an input and generate accurate output.

By analyzing the parameters of the model and the size of the dataset, BART-base is selected, which is considered to be a light and distilled version of the original BART. Thus, it is found to be capable of retaining most of BART, making it ideal for a real-time question-answering environment. BART-base is implemented in the project involving various phases, which include fine-tuning BART, training on the dataset, and answer generation.

BART-base, which is trained on a larger corpus of diverse text, is further trained on the dataset. The aim of the fine-tuning process is to adapt the model in such a way that it is capable of understanding computer science language and concepts. This process is enabled to provide the model with the ability to offer predictions with accuracy and reliability.

During the training of the model, different question-answer pairs related to specific topics of computer science are exposed to the model. The dataset is split into a training set (90%), a validation set (10%), and a test set containing altogether 20 question-answer pairs. The training process is enabled to help the model learn relevant domain-specific terminologies, grammatical structures, and contextual relationships. After the completion of the fine-tuning process, appropriate answers to new questions are generated by the model. The model is benefited from its learned computer science concepts to create new and relevant sentences.

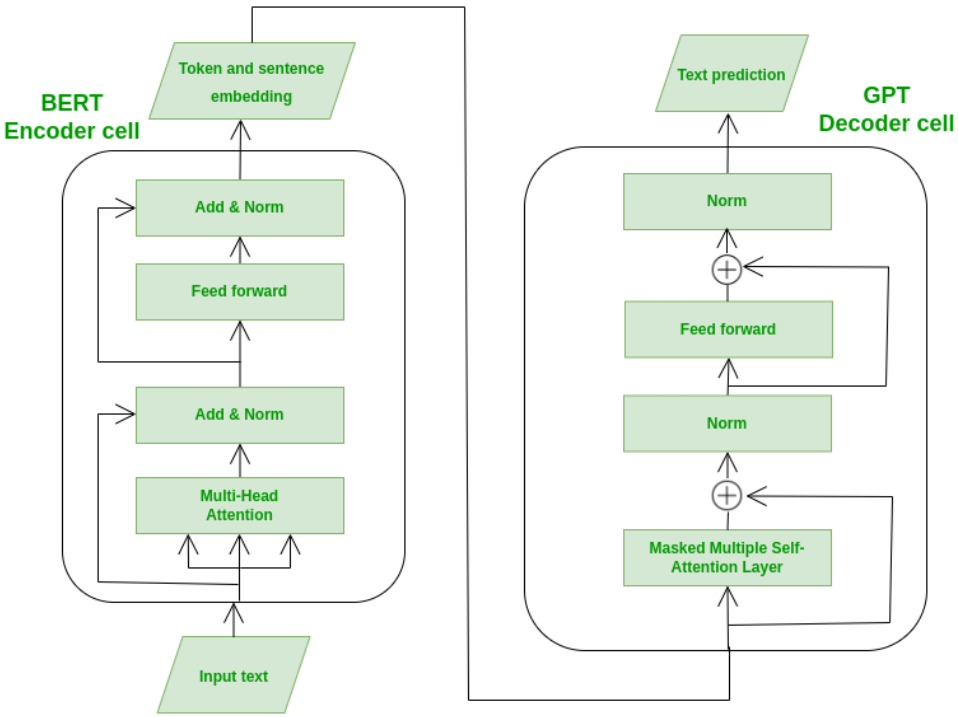

Figure 2: BART Architecture

Source: https://www.geeksforgeeks.org/bart-model-for-text-auto-completion-in-nlp/

3.1.3.2 Text-to-Text Transfer Transformer

T5 is described as a transformer model that is developed by Google Research, which treats NLP tasks as a text-to-text problem—meaning input and output are formatted as text strings. This approach is enabled to allow T5 to perform tasks like text classification, summarization, translation, and question answering using an encoder-decoder architecture. The encoder-decoder architecture of this model is made efficient for understanding complex questions and generating answers accurately.

By understanding the overall requirements of the project and analyzing the available resources, the T5-small variant is selected, which strikes the balance between performance and efficiency. T5-small, being a very light model, is pre-trained on large-scale text data and is fine-tuned using the dataset to convert it into a

desirable system. T5 is regarded as a very versatile model, as new text is synthesized as an answer instead of presenting the same text as in the dataset.

This model is used to process text with the help of a tokenizer called ‘SentencePiece’, which is employed to convert the text into suitable tokens as required by the T5 architecture. Each query is encoded and the answer is decoded during the process called forward pass, and then loss is calculated. The difference between the predicted and generated answer is measured with the help of a loss function. The errors encountered by the loss function are propagated back through the model via backpropagation while the model's weights are adjusted. The model’s understanding is enhanced through repeated exposure to question-answer pairs.

When a question is given by the user as input, it is processed by the T5 encoder, where token relationships are analyzed to understand meaning. The answer is given by the decoder token by token, ensuring accuracy. T5-small is found to excel in domain optimization via wide presentation on diverse text. The lesson reading paradigm is used to simplify question-answer assignments. With low parameters, strong performance is maintained by reducing calculation overheads. The transformer-based architecture is ensured to provide consistent, reference-conscious answers, making it ideal for abstract question-response tasks.

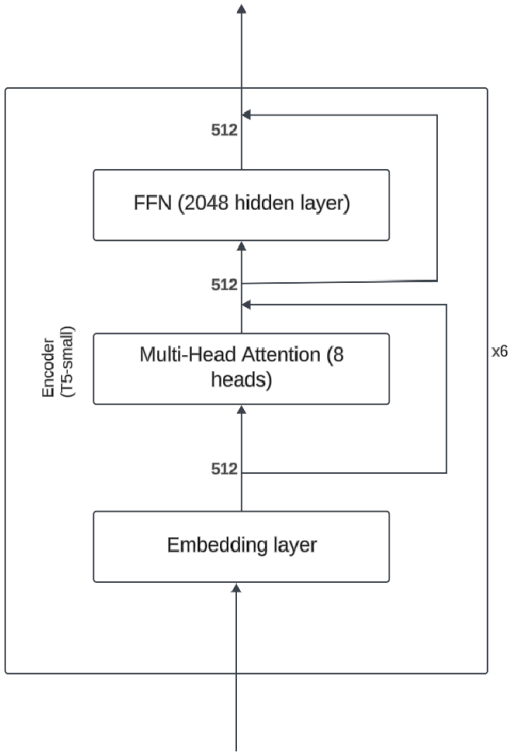

Figure 3: T5 Architecture
Source: https://www.researchgate.net/figure/T5-small-encoder-architectureref ig3381509245

4. Result and Analysis

4.1 Dataset Visualization

Table 1. Few samples of Dataset

Question	Answer
What are the prerequisites to learn Microsoft Azure?	To learn Microsoft Azure there is no need of any extra knowledge any one can easily learn the Azure if they have the below listed set of skills. Basic understanding of Azure concept. Understanding of Cloud Concepts Understanding if basic infrastructure management, database management, and software development.
How to install Docker on Windows?	If you are using Windows 10 Pro, you can install Docker directly. Otherwise, you will need to use Docker Toolbox, which runs Docker inside a virtual machine2 min read
What is a website?	A website is a collection of many web pages, and web pages are digital files that are written using HTML (HyperText Markup Language). To make your website available to every person in the world, it must be stored or hosted on a computer connected to the Internet round a clock. Such computers are know5 min read
What is a Dial Modem?	A Dial Modem is a type of modem that converts data used in telephone and data used on computers. In short, dial modems convert between analog form and digital form. The networking devices connected to the computer are all at one end, and the telephone line is at another end. This type of modem transmits data at a speed of 56,000 per second.
How can outliers be effectively removed from a dataset?	Removal involves identifying and excluding outliers based on statistical thresholds (e.g., z-scores, IQR) or machine learning models (e.g., Isolation Forest). Removing outliers may help reduce model variance but should be done carefully to avoid losing important information.

4.2 Dataset Diversity

Table 2. Data diversity and biasness

Subfield	Percentage
Software Engineering	17.67%
AI/ML	15.80%
Operating System	15.51%
Networking	14.09%
Theory	9.88%
Database	9.21%
Data Science	5.76%
Cloud Computing	4.20%
Human-Computer Interaction	3.50%
Computer Architecture	1.76%
Programming Language	1.61%
Security	1.01%

4.3 Observation

Initially, the model's performance is observed on questions that are provided without fine-tuning on the datasets. To evaluate the model's performance, the standard NLP metric ROUGE is used. The results are shown in Table 3 below.

Table 3. Model performance before fine tuning

Model	Question	Generated Answer	SCORE
T5-Small	What is artificial intelligence?	artificial intelligence	0.1739
BART-Base	What is artificial intelligence?	What is artificial intelligence?	0.1800

After the ROUGE scores are checked, it is observed that the model needs some tweaking to boost its performance. A variety of settings are experimented with to fine-tune how well the model performs. First, the learning rate, which basically dictates how much the model's internal settings, or "weights," get adjusted each time, is explored. A wide range of values is tested, and a learning rate of 5e-5 is found to be the sweet spot, striking the right balance between stable learning and achieving good results.

Next, the batch size, which refers to the chunk of training examples the model learns from before updating those internal settings, is examined. Batch sizes ranging between 2 and 32 are tested. A batch size of 8 is ultimately selected for training. This choice is considered effective in balancing computational efficiency with model performance, avoiding memory constraints while maintaining stable gradient updates. The smaller batch size is also found to contribute to better generalization compared to larger alternatives tested during hyperparameter optimization. Empirical observations confirm that this configuration yields optimal results within the available hardware limitations.

To begin, both models are trained for 5 epochs to assess how they perform relative to each other. A sample dataset from Hugging Face's library is used, with about 5,342 questions pulled for this initial test run.

Table 4. Model performance after fine tuning

Model	Question	Generated Answer	SCORE
T5-Small	What is artificial intelligence?	Artificial intelligence is a technique in artificial intelligence that enables AI to understand and understand the dynamics of data, enabling AI to gain insights and insights.	0.2492
BART-Base	What is artificial intelligence?	Artificial intelligence is an artificial intelligence that enables AI to understand the data and gain its insight. It is used for language translation, speech recognition, or language translation.	0.2563

Upon training with a dataset tailored to the specific domain, BART-Base is observed to be more adept at generating answers than T5-small. Because of this, the decision is made to continue training with the BART-Base model. Determining the appropriate number of epochs is considered very important to ensure the model's accuracy without encountering underfitting or overfitting. The final dataset, comprising 20,076 question-answer pairs, is divided as follows: 18,050 for training, 2,006 for validation, and 20 for testing. To identify the epoch that yields the best results, a method of iterative training and validation is used. After each epoch, a checkpoint is created to save the model, and its performance is evaluated on the validation set using the ROUGE score. If the ROUGE score shows improvement, the model is designated as the best_model and saved accordingly. Training is then continued to the next epoch.

As training progresses, the training loss steadily decreases and the average ROUGE score increases with each epoch. However, upon reaching the 22nd epoch, a drop in the ROUGE score is observed, suggesting that overfitting is beginning to occur. Because of this, the model is found to perform best at the 21st epoch, and this version is saved as the best_model, achieving an average ROUGE score of 0.2867.

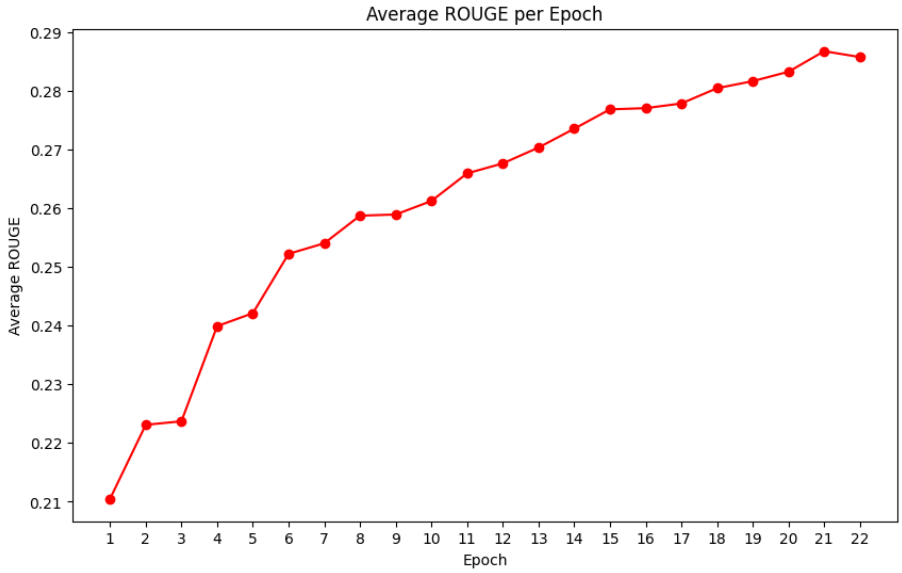

Figure 4: Average ROUGE per Epoch

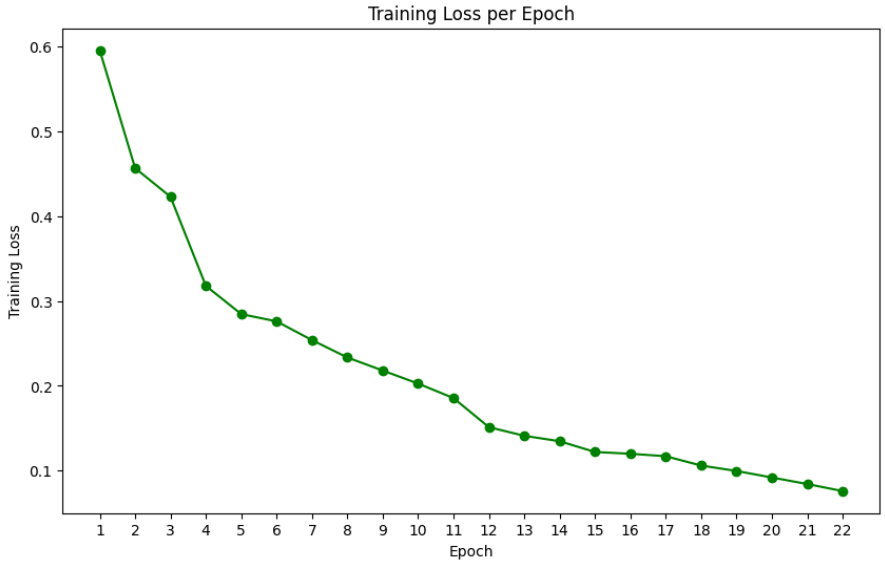

Figure 5: Training Loss per Epoch

The final evaluation is conducted using the test dataset, consisting of 20 question-answer pairs, which had been separated during the initial data splitting phase. Upon testing, the fine-tuned BART-Base model achieves an average ROUGE score of 0.2890, demonstrating its effectiveness in generating accurate and relevant answers for computer science-related questions. This result is confirmed to show that the model generalizes well to unseen data while maintaining performance close to its validation-phase benchmark (ROUGE-21: 0.2867).

Table 5. Model performance on Test data

Question	Reference Answer	Generated Answer
What is Gradle?	Gradle is an excellent open-source construction tool for software development, developed by Hans Dockter and others.	Gradle is an open-source Gradle tool that allows you to automate building, testing, and deployment of Gradle models. It provides tools for Gradle editors to build, test, and deploy models efficiently.
Can you explain the concept of encapsulation in object-oriented programming?	Encapsulation is the concept of bundling data and functions (methods) together in a single unit known as a class in object-oriented programming. It allows for better organization and reusability of code by encapsulating data and functionality within a single entity.	Encapsulation in object-oriented programming refers to the practice of hiding the implementation details and only showing the essential functionality or features to the user. This helps simplify code and improves modularity, especially in large codebases.

Table 6. Some random questions with their answers

Question	Generated Answer
What is artificial intelligence?	Artificial intelligence refers to systems that can perform tasks that typically require human intelligence, such as language translation, speech recognition, or facial recognition.
Define DBMS.	Database Management System (DBMS) refers to the system used to store, organize, and manage data in a database. It ensures data integrity, consistency, and efficient access to resources, acting as the interface between users and the system.
Explain the services provided by AWS	The following are the services provided by AWS: Data Storage and Databases: Amazon S3 provides structured data storage and retrieval for unstructured or semi-structured data. Amazon EBS offers automatic backups and support across multiple Availability Zones. Amazon Cloud Platform (AWS): AWS offers Internet of Things (IoT) services, including IAM, VPC, and EKS for managing and accessing cloud resources.
Define transformer.	A transformer is a neural network architecture consisting of an encoder and a decoder. The decoder processes input sequences, while the decoder generates output sequences through self-attention. They're widely used in tasks like text summarization, speech recognition, and machine translation.
Which is the most popular programming language?	The most popular programming language in the world is Python. Python is a versatile and easy-to-learn programming language that is widely used for various applications, including web development, data analysis, machine learning, and more. It has a vast ecosystem of libraries and frameworks that make it easy to learn and use it.

4.4 Comparative Performance Analysis

Table 7. Performance comparison of TechBot with existing domain-specific QA systems

System	Domain	ROUGE	Human Rating
TechBot (BART)	Computer Science	0.2867	4.1/5.0
CodeQA (Chen, 2022)	Computer Science	0.251	3.8/5.0
KI-MAG (A. Ekbal, 2024)	Medical	0.267	3.5/5.0

5. Conclusion and Future Enhancements

To boost the model's capabilities even further, the focus can be placed on broadening its knowledge base. This involves bringing in more data from diverse sources like academic research, technical manuals, and coding forums. These valuable datasets can be sourced from public repositories, collaborations with universities, and partnerships with industry leaders. The model can also be provided with a more well-rounded education by utilizing information from different areas like math, physics, and engineering. Specifically, resources like the Mathematics Dataset (e.g., GSM8K), physics problem collections, and technical documents can be leveraged. All of this would help the model become more versatile and better at handling a wider array of subjects.

In addition, implementing Chain-of-Thought (CoT) reasoning can be prioritized, leveraging GPT-3.5 to generate stepwise explanations that mimic human problem-solving. This framework would break complex problems into smaller, interpretable steps, validated through user studies to ensure clarity and accuracy. To enhance the model's reasoning capabilities, it could be trained on structured datasets like AQuA (algebra), StrategyQA (logical deduction), and expanded datasets from partnerships with platforms like Coursera and edX, granting access to 50,000+ tutorials for diverse, high-quality training data.

Furthermore, user feedback mechanisms where users rate answer quality could be integrated to refine the model iteratively. These improvements would not only boost task-specific performance but also broaden the model's applicability in education, research, and industry. By strengthening its reasoning and expanding its knowledge base, the model could better assist students grappling with complex problems and professionals requiring in-depth technical insights.

Acknowledgements

We express our gratitude towards the Department of Electronics and Computer Engineering, Kantipur Engineering College, for helping us conduct the research as an academic project. We convey our profound appreciation to our project supervisor Er. Ajay Mani Paudel for providing us with his support and the timely checks of our models.

We would also like to thank our teacher Er. Pralhad Chapagain, senior Lecturer/ Acting Chief, Research Training and Consultancy Division (RTCD) for immense support in writing this research paper and making it possible.

References

A. Ekbal, K. M. a. A. Z., 2024. KI-MAG: Knowledge-Infused Abstractive Question-Answering System for Medical Applications. *arXiv preprint,* pp. 1-14.

A. Vaswani, N. S. N. P. J. U. L. J. A. N. G. Ł. K. a. I. P., 2017. Attention is all you need. *in Advances in Neural Information Processing Systems,* pp. 5998-6008.

A. Zafar, K. M. a. A. E., 2024. MedLogic-AQA: Enhancing Medical Question Answering with Abstractive Models Focusing on Logical Structures. *arXiv preprint,* pp. 1-14.

C. Raffel, N. S. A. R. K. L. S. N. M. M. Y. Z. W. L. a. P. J. L., 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.,* pp. 1-67.

Chen, L. e. a., 2022. CodeQA: A Question Answering System for Programming Tutorials. *ACM SIGIR.*

J. J. Bird, A. E. a. D. R. F., 2024. Chatbot Interaction with Artificial Intelligence: Human Data Augmentation with T5 and Language Transformer Ensemble for Text Classification. *arXiv preprint,* pp. 1-18.

J. Liu, D. S. K. L. a. J. Z., 2024. Graph-Based Knowledge Integration for Question Answering over Dialogue. *arXiv preprint,* pp. 1-14.

Lin, C.-Y., 2004. Rouge: A package for automatic evaluation of summaries. *in Proceedings of the Workshop on Text Summarization Branches Out.*

M. Lewis, Y. L. N. G. M. G. A. M. O. L. V. S. a. L. Z., 2020. BART: Denoising Sequence-to-Sequence Pre-Training for Natural Language Generation, Translation, and Comprehension. *58th Annu. Meeting Assoc. Comput. Linguist. (ACL),* pp. 7871-7880.

Rekabdar, J. J. a. E., 2020. Enhancing Knowledge Base Question Answering Using BERT and Attention Mechanisms. *arXiv preprint,* pp. 1-12.

Smith, J. &. L. K., 2023. Domain-Specific QA with BERT in Software Engineering. *IEEE TSE.*