CodeCraft: Leveragin
Automated Code G

Sumedh Ghavat, Tanmay Arma

Introduction

In the realm of Natural Language Processing (NLP), the ability to generate coherent and effective code has long been a coveted pursuit. Imagine a system that, when presented with a problem statement, can not only

Results

1. BLEU and CodeBLEU Sc
   - Easy Problems:
     - BLEU: 0.85, Cod
     - High accuracy a
   - Medium Problems:
     - BLEU: 0.65, Cod

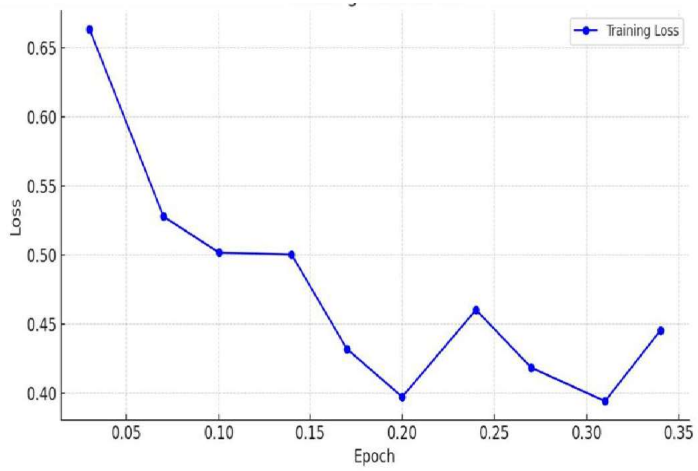
ng LLMs for
eneration

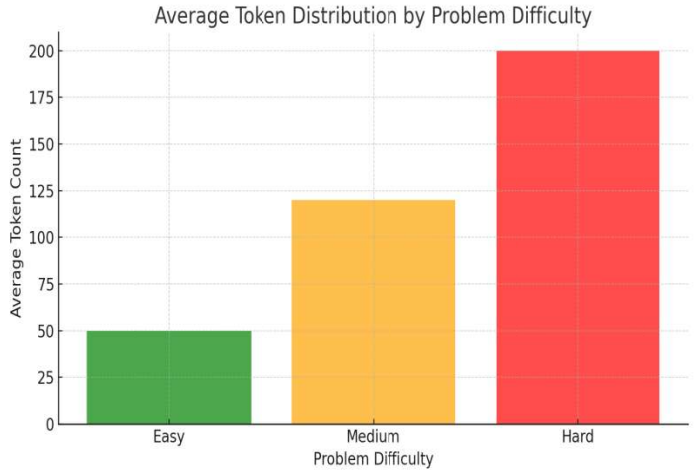
al, Shreyas Habade

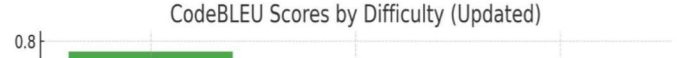
cores:

eBLEU: 0.78
nd coherence.

eBLEU: 0.60







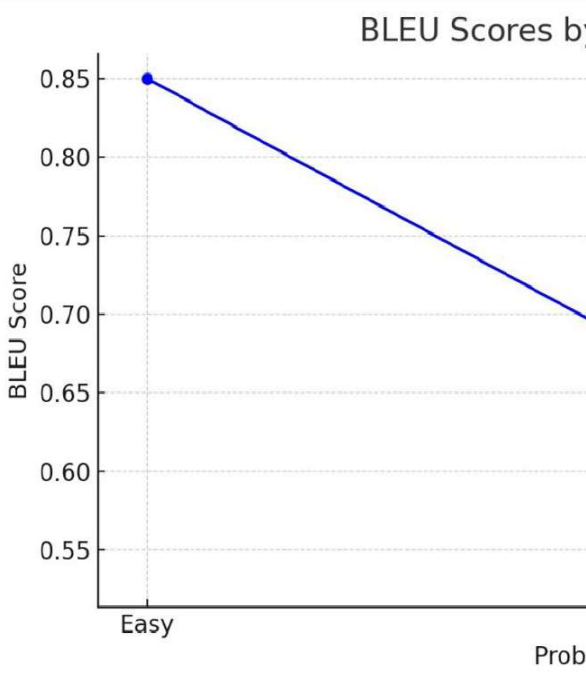
comprehend the task at hand but also produce high-quality solutions akin to those crafted by seasoned programmers.

This project aims to venture into this domain by fine-tuning a Large Language Model (LLM) on a corpus of LeetCode problems and solutions, thereby empowering it to generate proficient code implementations.

Background

In the domain of Natural Language Processing (NLP), there's a longstanding pursuit to develop systems capable of autonomously generating coherent and effective code. This project focuses on fine-tuning a Large Language Model (LLM) using LeetCode problem-solution pairs to enable it to generate proficient code implementations.

- Satisfactory per
- Hard Problems:
  - BLEU: 0.45, Cod
  - Challenges in ha



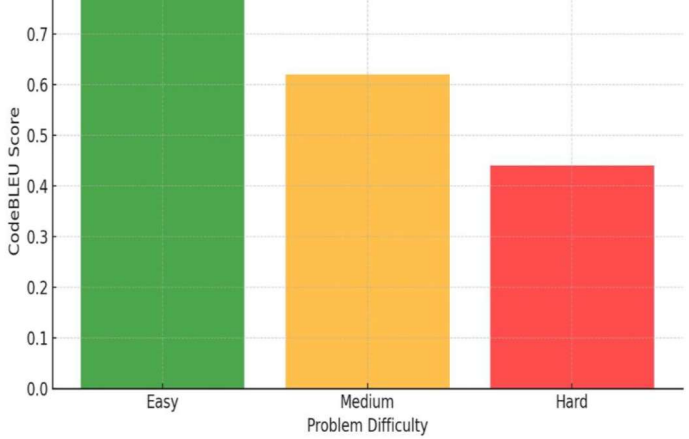
Method

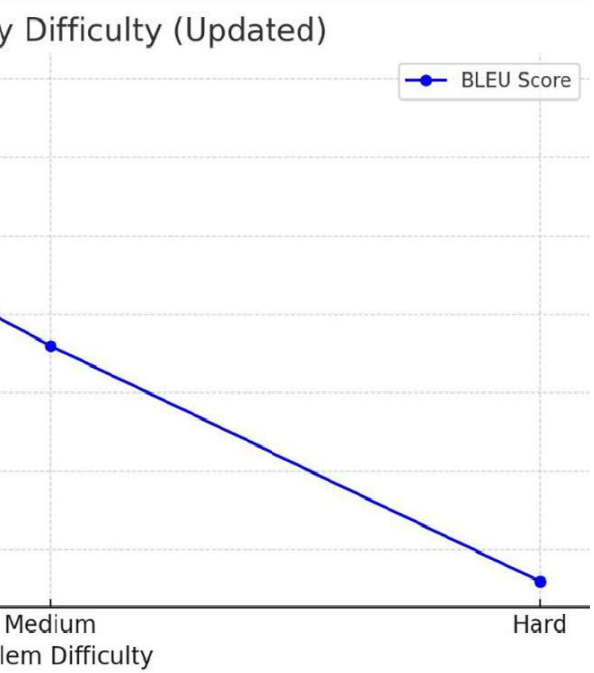
1. **Model and Data S**
- Selected the Code
- Utilized the greeng
- Transformed the
  Python solutions fo

formance with occasional lapses.

eBLEU: 0.40

andling complex requirements.





1. Insights:
   - Strengths:
     - Proficiency in solving simpler tasks.
     - Accurate solutions for easy problems.
   - Challenges:
     - Difficulties with nuanced logic.
     - Limitations in handling edge cases.
2. Graphical Representation:
   - BLEU Score Trend:
     - Decreasing trend with increasing complexity.
   - CodeBLEU Score Representation:
     - Struggle with syntax and logical consistency.

election:

Llama 2 model specialized for code generation tasks.

erong/leetcode dataset, comprising LeetCode problems and solutions.

dataset into a Pandas DataFrame, merging problem statements and r easier manipulation.

Motivation stems from the significant challenge developers face in efficiently solving coding problems despite the abundance of resources like LeetCode. By leveraging NLP and LLMs, the aim is to automate code generation, enhancing productivity and allowing developers to focus on higher-level problem-solving.

Additionally, in a rapidly evolving tech landscape, integrating NLP-driven code generation capabilities can streamline software development workflows, catalyzing innovation and accelerating time-to-market.

Data

For this project, we utilized the greengerong/leetcode dataset from Hugging Face, comprising programming problems and solutions sourced from LeetCode. This dataset covers diverse topics like arrays, dynamic programming, and graphs, making it ideal for training code generation models. It includes structured pairs of problems and solutions, along with metadata indicating problem

2. **Tokenization and**
- Employed the Auto
- Configured pad_tok
- Loaded the model
quantization to enh
- Important configura
and Double Quanti

3. **PEFT (Parameter-**
- Integrated the peft
- Utilized LoRA to tra
- Configured LoRA b

Conclusion

The motivation behind th
efficiently solving coding
LeetCode offering a pleth
remains labor-intensive a
alleviate this burden by a
enabling developers to fo
syntax

Model Configuration:

Tokenizer from Hugging Face Transformers to tokenize input data.
ken to match eos_token for consistent padding during training.
with AutoModelForCausalLM, implementing BitsAndBytesConfig for 4-bit
ance memory usage and performance.
ations included Quantization Type (nf4), Compute Data Type (float16),
zation.

Efficient Fine-Tuning):

library for efficient training using Low-Rank Adaptation (LoRA).
ain specific model layers, speeding up fine-tuning with reduced data.
y setting the r Parameter to determine the rank of the low-rank matrix.

his endeavor stems from the perennial challenge faced by developers in
g problems. Despite the abundance of resources and communities like
ora of problems and solutions, the process of crafting optimal code solutions
nd time-consuming. By leveraging the power of NLP and LLMs, we aim to
utomating the code generation process, thereby enhancing productivity and
ocus more on higher-level problem-solving tasks rather than the minutiae of
and implementation details.

acterized by rapid technological advancements and an ever-growing demand

difficulty (Easy, Medium, Hard). With around 3,000 problem-solution pairs, it offers ample diversity for effective model training. Each entry includes problem descriptions, sample input/output, topic tags, and optimized Python solutions, enhancing the models' generalization capabilities.

Result

Difficulty	BLEU	CodeBLEU
Easy	0.85	0.78
Medium	0.65	0.60
Hard	0.45	0.40

Table: Summary of results across easy, medium and hard problems

Moreover, in an era chara
for software solutions, the
importance. By integrati
workflows, we can pote
software applications,

References

1. OpenAI, "OpenAI Codex: Po
   codex/
2. MDPI Editorial, "Google Alp
3. ACL Anthology, "CodeT5+:
   for Computational Linguistic
4. Papers With Code, "GPT-4 a
5. Hugging Face, "Datasets: A
   https://huggingface.co/datas
6. LeetCode, "Explore Coding
7. G. Dey and A. Ganesan, "In
   10, no. 3, pp. 123-134, 2023
8. Gupta et al., "The Impact of

actenized by rapid technological advancements and an ever-growing demand
e ability to expedite the software development lifecycle assumes paramount
ing NLP-driven code generation capabilities into existing development
ntially streamline the process of prototyping, debugging, and deploying
thereby catalyzing innovation and accelerating time-to-market.

owering GitHub Copilot," *OpenAI Blog*, 2021. [Online]. Available: https://blog.openai.com/openai-

haCode: Competing with the Coding Elite," *MDPI Journals*, 2022.
Enhancing Code Generation Models," *Proceedings of the 59th Annual Meeting of the Association s*, 2021.
and In-Context Learning for Code Generation," *Papers with Code*, 2023.
Hub of Public Datasets for Machine Learning," *Hugging Face*. [Online]. Available: sets
Challenges and Solutions," *LeetCode*. [Online]. Available: https://leetcode.com
struction-Tuned LLMs for Social Scientific Tasks," *Journal of Artificial Intelligence Research*, vol. 3.
Instruction Tuning on Large Language Models," *NLP Conferences*, 2023.