Akshay Verma
DATS 6312 - Natural Language Processing for Data Science
Final Project - Individual Report
12/11/2023

Introduction

Our project aimed to use transformers to answer math word problems, this task can be seen as a form of machine language translation where the target language is mathematical equations and numerical answers. After some experimentation, we decided to use the T5 model by Google for this task. The shared work was the experimentation required to reach the T5 model, finding the data, and work on the Streamlit app and report. The T5 small model used for numerical outputs was coded by Paul, and the hyperparameter tuning and the graphs for validation and training loss were done by Paul aswell. Carrie worked on the earlier renditions of our project with the GPT and BERT models, and she also headed the work on the report. Jack worked on the Streamlit app and helped the rest of the team members with their models.

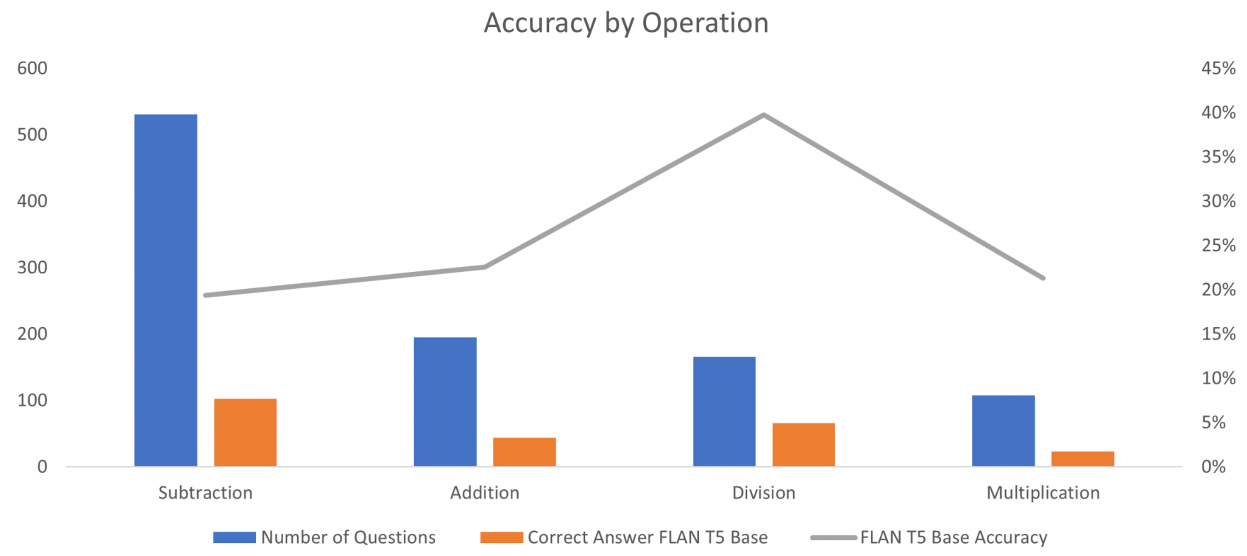
Description of Individual Work

At first, I worked on coding a transformer from scratch and training it on our dataset. That endeavor proved fruitless as the model could not handle out-of-vocabulary tokens. Meanwhile, Paul was working with T5 Small and reached some fraction of success in the predictions. So, I switched to T5 architecture as well for equation generation. At first, I tried tuning the T5 small and was able to get equations as predictions but the accuracy and ROUGE scores were very low. I then experimented with different T5 versions and decided upon the Flan T5 base as it had the best tradeoff between model size, computational cost, and accuracy.
I later experimented with different models, like LLAMA 3B tuned on GSM8k with QLoRA but the computational cost was too much and the predictions while more accurate were not equations.
After deciding on the Flan T5 Base, I created a test script to test the model on our testing dataset and get our metrics - Which I later visualized through Excel. Then I created a script to upload our models to HuggingFace and then worked on the Streamlit app and connected it to our HuggingFace models for real-time inference.

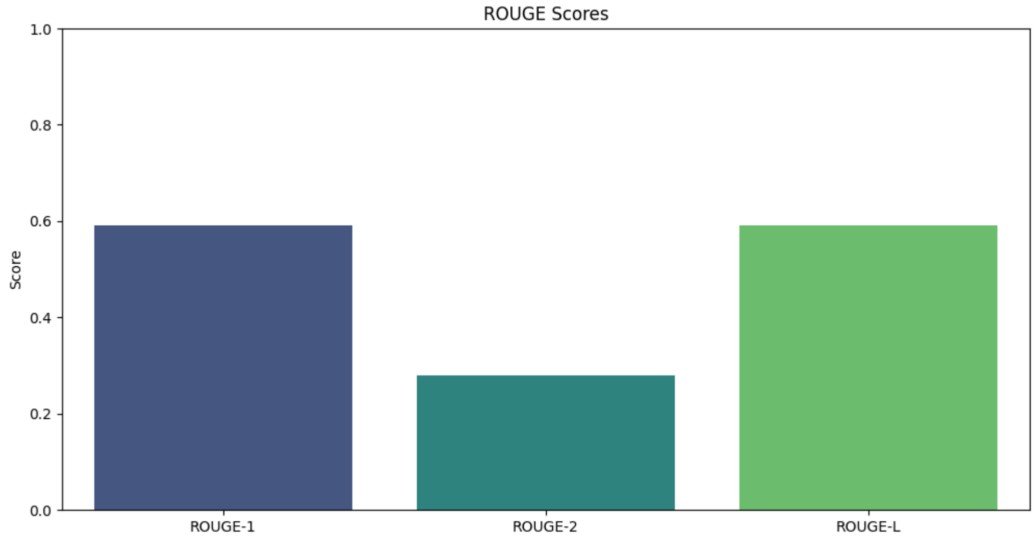
Results:

So our Flan T5 Base model achieved an accuracy of 22% for equation generation but the accuracy did vary by operation. The division was the highest nearing 40% accuracy whereas

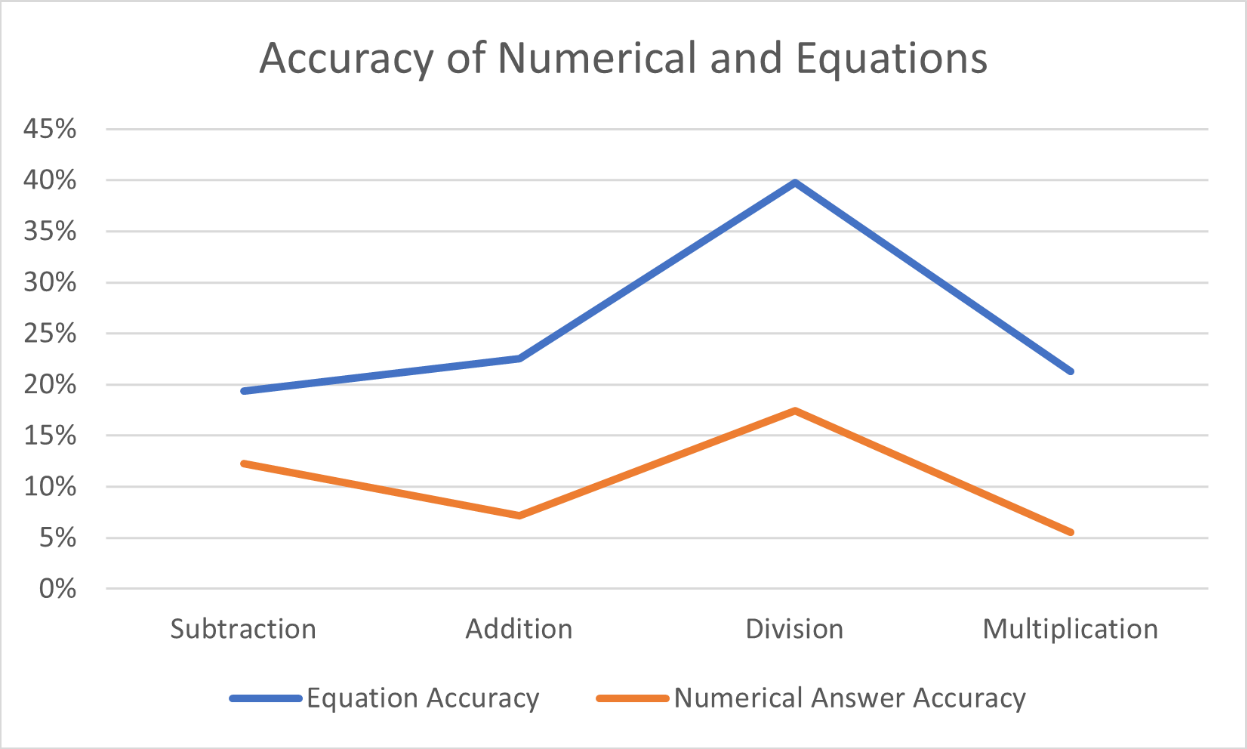
Multiplication was the lowest with just below 20% accuracy. I used Sympify, which is a python function to evaluate the answers to the generated equation.



For ROUGE scores we got an f score of 0.6 for ROUGE 1 which just checks the similarity between monograms, for ROUGE 2 we got an f score of nearly 0.2.



Our Equations generation model has much higher accuracy between all operations than the model that generates numerical answers.



Summary and conclusions

This Project underscores how language models still struggle with mathematical problems. This is also true with state-of-the-art language models like GPT 4 and Gemini which have only achieved accuracy of around 50% with the MATH dataset. We were able to achieve an accuracy of 23.6% with our T5 model. I learned a lot about different kinds of transformers as we went through decoder-only models and encoder-only models to finally reach an encoder-decoder model that was finally able to do the task we wanted.

For future improvements, we can use a larger model to get better accuracy. We can also tune the model on a different mathematical dataset like MATH, GSM8K, etc to get a more versatile model.

Original Code Percentage

Around 60% of the code I used was either original or modified.

Reference

Doe, P. (2021). Rouge Your NLP Results. Medium. https://medium.com/@priyankads/rouge-your-nlp-results-b2feba61053a

Koncel-Kedziorski, R., Roy, S., Amini, A., Kushman, N., & Hajishirzi, H. (2016, June). MAWPS: A math word problem repository. In Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies (pp. 1152-1157).

Hugging Face. (2023). T5-small: Text-To-Text Transfer Transformer. Hugging Face. https://huggingface.co/t5-small

Hugging Face. (2023). T5-Flan-Base: Text-To-Text Transfer Transformer. Hugging Face. https://huggingface.co/google/flan-t5-base

Jacob, John. (2023). What is FLAN-T5? Exemplary AI Blog. https://exemplary.ai/blog/flan-t5

Patel, A. (2022). SVAMP: Structural Variant Annotation, Mapping, and Prediction. GitHub. https://github.com/arkilpatel/SVAMP

Patel, A., Bhattamishra, S., & Goyal, N. (2021). Are NLP models really able to solve simple math word problems?. arXiv preprint arXiv:2103.07191.

MathBot. (2023). MAWPS_Augmented.pkl. GitHub. https://github.com/Starscream-11813/MathBot/blob/main/MAWPS_Augmented.pkl

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res., 21(140), 1-67.