Achieving Machine Reasoning for Math Problems Through Step-By-Step Supervision Signal

Brian Do, Phillip Hoovestal, Max Sobol Mark
{brianhdo, philhoov, joelmax}@stanford.edu

OVERVIEW

- Multi-step reasoning problems are hard for machine learning due to the non-smooth loss functions associated with them. General math problems fall in this category
- Education studies show that humans learn math by doing example problems and comparing intermediate steps
- **OUR GOAL**: By **supplying intermediate steps** as additional labels, we aim to **shape the loss function** and **improve accuracy**

DATASET

We used a **mathematics dataset containing text question and answer pairs** on a variety of mathematics topics **released by Google's Deepmind** in 2019. An example question-answer pair is:

Q: *Find the second derivative of q^5 - 391*q^4 + 1600*q^2*
A: *20*q^3 - 4692*q^2 + 3200*

We modified the dataset by supplying intermediate steps as additional labels for amenable problem types which gave the Transformer model difficulty in Google's DeepMind paper. An example modified question-answer pair is:

Q: *Calculate (-168)/(-2) + (28 - 74).*
A: *168 / 2 + (28 + -74)||84 + (28 + -74)||84 + -46||38||*

where '||' denotes the end of a step

METHODS

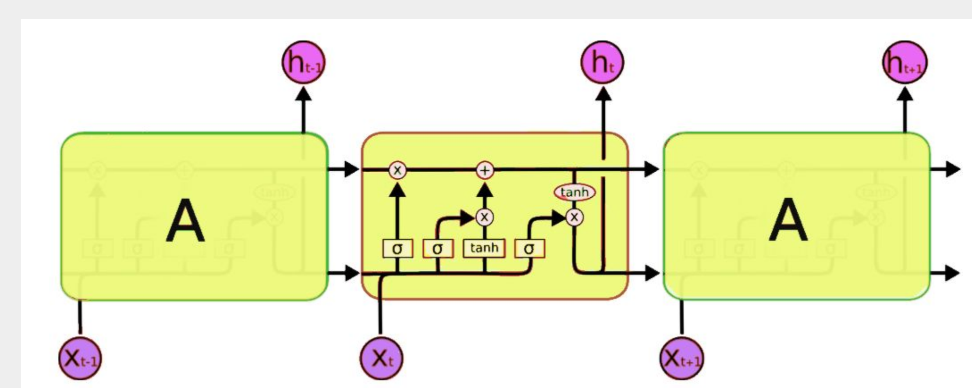
Linear Regression

As a simple baseline, we used linear regression with a **polynomial based kernel** along with **L2 regularization** to reduce overfitting

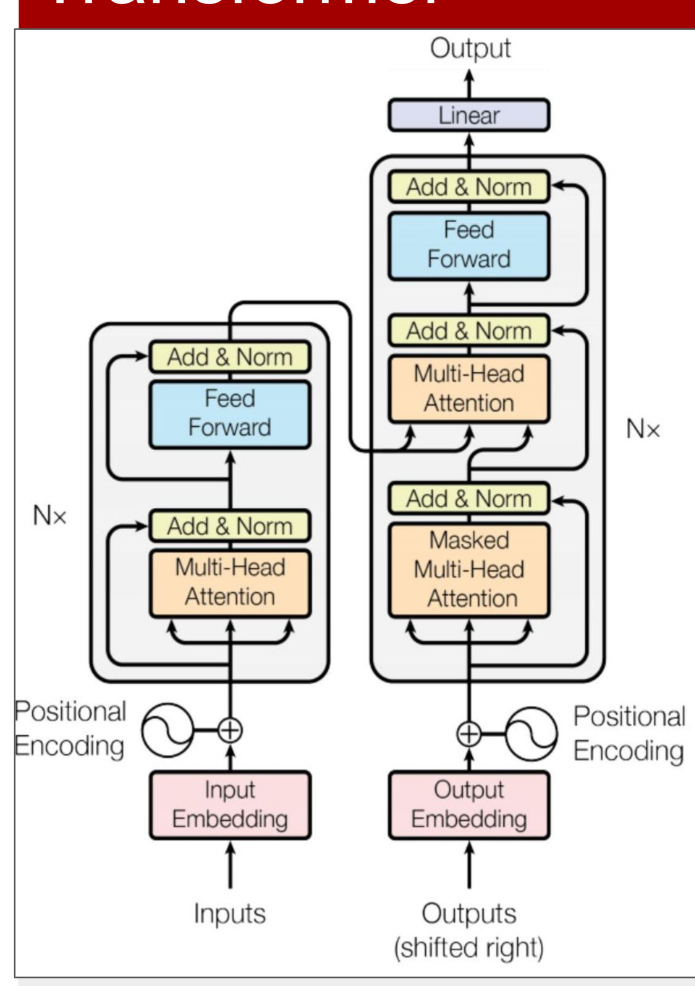
Dataset was modified to allow for use by linear regression:
- Complex questions requiring linguistic parsing were omitted
- Problems were converted from strings to vectors
- Question lengths were fixed

Long Short-Term Memory (LSTM)

LSTM is a type of recurrent neural network (RNN) which has feedback connections and can handle sequential data. We thus used sequence-to-sequence LSTMs with and without attention as more complex baselines



Transformer



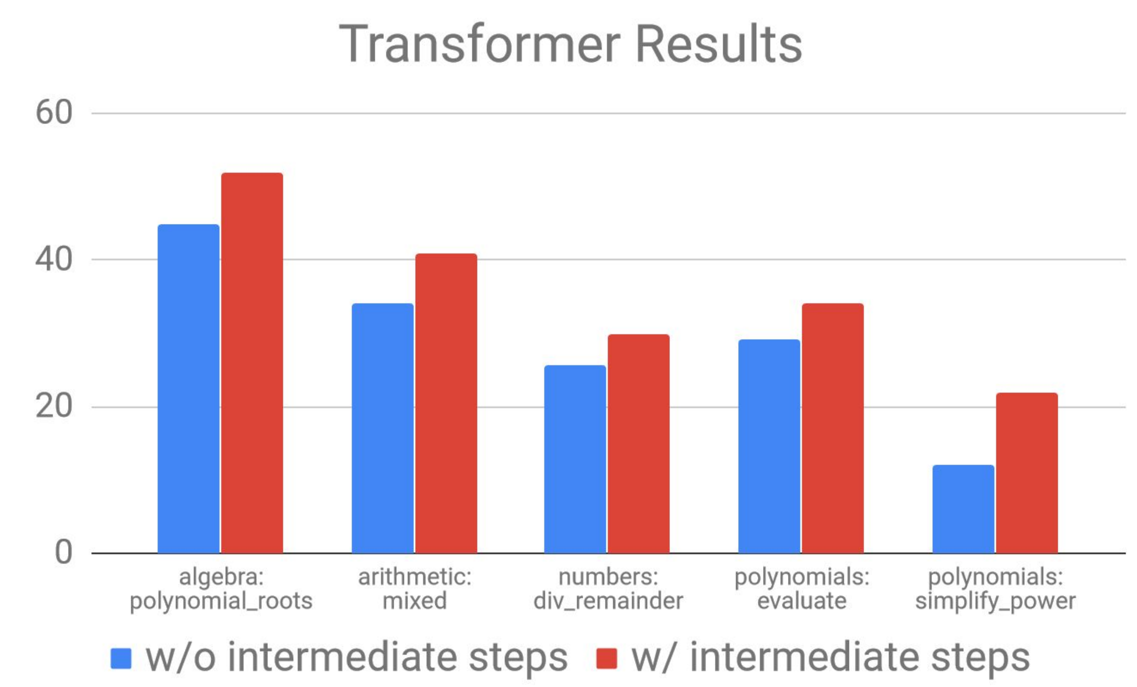
The Transformer model's key feature is a self-attention mechanism, allowing it to better retain relationships between distant characters than LSTMs.

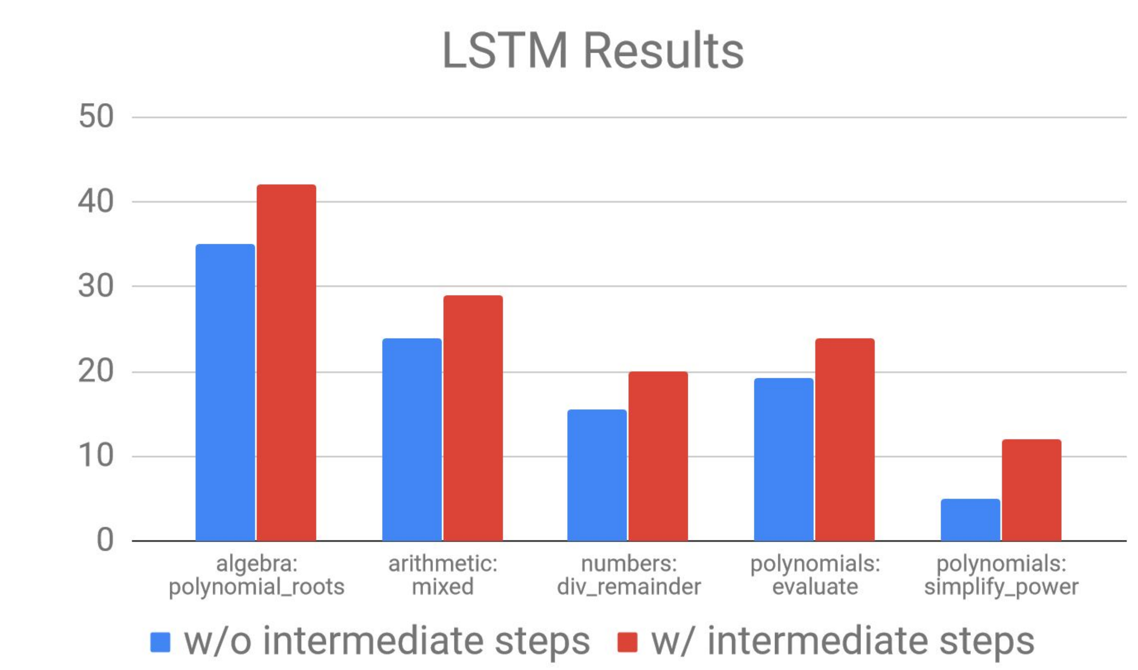
Because of this, the Transformer often has more success parsing complex phrases than an LSTM, as indicated by DeepMind's results

RESULTS

Linear regression:	Addition Only	Multiplication Only	Mixed Arithmetic
No intermediate steps	100%	100%	0.7%
With intermediate steps	100%	100%	1.5%

Linear regression is too simple for these types of problems. **Intermediate steps have no effect**





Including intermediate steps does result in a **slight improvement in** training **accuracy** for the Transformer model and the LSTM model, likely due to **implicit shaping of the reward function** to include the steps necessary to get correct final answers

NOTE: Transformer and LSTM results are **not** finalized! We are still working on training w/ intermediate steps

FUTURE WORK

Perform grid search for hyperparameters using additional computational resources for LSTM and Transformer models to better understand what types of architectures best benefit from intermediate steps