**MathBot – A Deep Learning Based Elementary School Math Word Problem Solver**

{ Anish Kumar Nayak, Rajeev Patwari, Viswanathan Subramanian }  {anishkn, rpatwari, vsub}@stanford.edu

**Motivation**

- Application of any learning algorithm to reduce natural language based math problems into equations is a topic of recent research
- Success of techniques such as Deep NLP, RNN flavors, Transformers etc. in this area to form a milestone towards general artificial intelligence
- Eventually build an end-to-end application, to assist elementary school parents and teachers

**Datasets**

- Source: MaWPS, Dolphin18k, Alg514, Draw
- Preprocessing done to extract, sanitize and number map dataset
- Each data point contains input sequence (problem), output sequence (equation), and final solution

**References**


**Bi-LSTM Encoder, LSTM Decoder Attention Model for Initial Setup**

**Transformer Model**

**Loss Function & Accuracy Metrics**

- SparseCategoricalCrossentropy
- SparseCategoricalCrossentropy, Eqn Solver

**Results**

**Plots using Tensorboard data generated for Model accuracy and loss on validation set**

**Table 2: BLEU scores with baseline Bi-LSTM, LSTM Attn. Model on MaWPS dataset (Embed Size, Hidden Size, Dropout)**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>BLEU-4</th>
<th>Solution ACC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ext-Elem</td>
<td>64.73</td>
<td>57.82</td>
</tr>
<tr>
<td>Combined-Elem</td>
<td>64.73</td>
<td>57.82</td>
</tr>
</tbody>
</table>

**Table 3: Cumulative BLEU & Solution accuracy scores with baseline Transformer Model(With batch size of 64, dropout of 0.1, and a custom learning rate schedule)**

**Hyperparameter Turning and Improvements**

- **Architectures & Hyperparameters Tuning**
  - Number of Layers: 4, 6, 8
  - Embedding Size: 54, 128, 256, 512
  - Hidden Size: 128, 256, 512, 1024, 2048
  - Number of Attention Heads: 8, 16, 32
  - Dropout: 0.0, 0.1, 0.15, 0.2, 0.5
  - Batch Size: 32, 64, 128
  - Learning Rate: 0.0005 through 0.1, and CustomSchedule

**Number Mapping for Word Embedding**

**Future Research**

- Implement beam search for transformer model to improve prediction quality
- Try larger AQUA-RAT dataset to extract equations, and obtain results
- Use transformer-XL and BERT with appropriate modifications
- Research on how to generalize to entirely new problem sets

**Discussion & Conclusion**

- Reproduced Bi-LSTM, LSTM Attn Model based work\(^1\) for initial setup
- Analyzed BLEU scores and predicted equations to conclude that BLEU score alone is not sufficient evaluation metrics
- Developed our equation solver, and used it to compute solution accuracy scores
- Further error analysis on baseline transformer results helped us to develop number mapping technique
- Using number mapping for word embedding, we obtained much improved results
- Dataset with elementary problems in general gave better results since they were cleaner
- Combined dataset scores were lower, since several examples had inconsistencies in problems and equations especially in Dolphin18k dataset
- Tuned transformer with number mapping for word embedding, and eqn solver acc metrics resulted in improved prediction

**Table 4: Table of BLEU and Solution Accuracy**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>BLEU-4</th>
<th>Solution ACC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ext-Elem</td>
<td>64.73</td>
<td>57.82</td>
</tr>
<tr>
<td>Combined-Elem</td>
<td>64.73</td>
<td>57.82</td>
</tr>
</tbody>
</table>

**Table 1: Composition of Datasets after Processing**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th>Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaWPS-Full</td>
<td>2965</td>
<td>100</td>
</tr>
<tr>
<td>MaWPS-Elem</td>
<td>1811</td>
<td>100</td>
</tr>
<tr>
<td>Ext-Elem</td>
<td>2107</td>
<td>250</td>
</tr>
<tr>
<td>Ext-Elem-Mapped</td>
<td>2107</td>
<td>250</td>
</tr>
<tr>
<td>Combined</td>
<td>9568</td>
<td>1000</td>
</tr>
<tr>
<td>Combined-Mapped</td>
<td>9568</td>
<td>1000</td>
</tr>
</tbody>
</table>

**Attention Function**

\[
\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}V\right)
\]

MultiHead\((Q, K, V) = \text{Concat}\{\text{head}_1, ..., \text{head}_n\}W^O\)

where, \(\text{head}_i = \text{Attention}(QW^Q, KW^K, VW^V)\)

**Equations**

- \(x = 696 + 109\)
- \(\text{Number of Seashells Benny has now}\)

**Word Problem**

Benny found 696 seashells and 109 starfish on the beach. He gave 248 of the seashells to Sally. How many seashells does Benny have now?