Machine Translation with `seq2seq`, Beam Search, and CopyNet

Jorge Avila
A14226029
University of California, San Diego
j3avila@ucsd.edu
email@domain

Abstract

Standard seq2seq models can be further improved by exploiting the characteristics of the architecture. These methods include attention, copying, and beam search which are the methods explored in this paper. We start training and analyzing a base seq2seq model then augmenting it with beam search and a CopyNet decoder to analyze the new performance characteristics. Unfortunately, in this case the base model was difficult to outperform with beam search model but the CopyNet model showed a fairly significant improvement in performance.

1 Machine Translation

The task of Machine Translation requires that input text from language be translated to another. `seq2seq` models are Encoder-Decoder models that incorporate bidirectional training, attention mechanisms, and word embeddings to capture context in an input sequence. Typically, both the encoder and docoder consist of RNNs, LSTMs, or GRUs as is commonly used in NLP tasks.

1.1 Beam Search

Beam search is a common method for finding the best output among k different output sequences from a model such as `seq2seq`. First, k^2 outputs are sampled from the model, k for each beam, then among them the k most likely are retained for the next token. These output sequences are maintained along with the hidden states and a running sum of their respective log probabilities. Summing the log probabilities is equivalent to computing the conditional probabilities of a token given the context. Beam searched is used in place of the common greedy search for the best sequence, where the token with the highest probability is chosen at each time step.

$$p(y_t|\mathbf{s}_t, y_{t-1}, \mathbf{c}_t, \mathbf{M}) = p(y_t, \mathsf{g}|\mathbf{s}_t, y_{t-1}, \mathbf{c}_t, \mathbf{M}) + p(y_t, \mathsf{c}|\mathbf{s}_t, y_{t-1}, \mathbf{c}_t, \mathbf{M})$$

Figure 1: Scoring function for CopyNet (Gu et al., 2016)

1.2 CopyNet

CopyNet is a model that occupies the decoder portion of the `seq2seq` model and it allows the model to predict out-of-vocabulary words. It does this by keeping track of two probabilities to create a ”mixture” of probabilities 1. These probabilities share a normaliziation term that allows them to compete via a softmax function. The first probability is the usual scoring function in a `seq2seq` model. The other is a probability that takes as input the encoder hidden state and encoded source sequence. It then outputs a distribution over the source sentence to determine what tokens to copy. (Gu et al., 2016), (Walsh, 2019)

One of the most important components of CopyNet is the attention mechanisms. CopyNet utilizes both **attentive read** and **selective read**. Attentive read is the usual attention mechanism used in `seq2seq` models. Selective read is a weighted sum over the encoded source states, just like attentive read, but the weights are the corresponding copy probabilities from the previous time step. Therefore, it is important to keep track of which tokens were copied.

2 Approach

2.1 Datasets

The only dataset used for experiments was the multi30k dataset with german to english translations (Elliott et al., 2016).

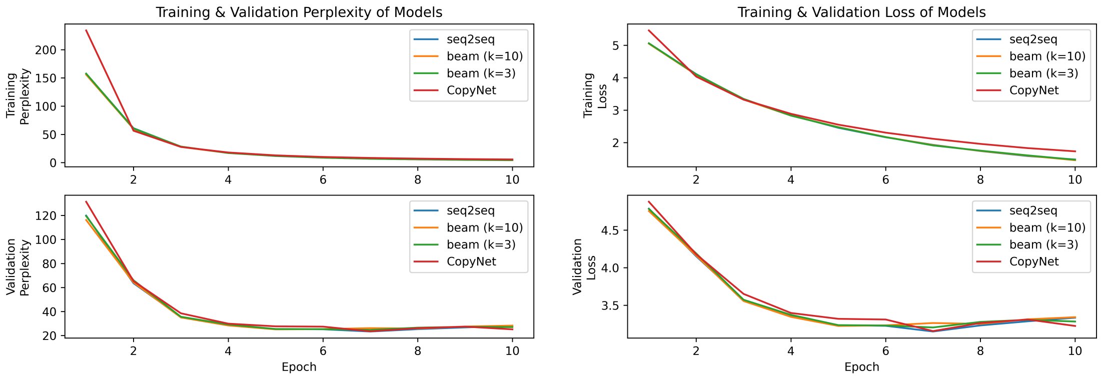

Figure 2: Training and Validation results for loss and perplexity. Validation results are on the bottom row.

2.2 seq2seq

The seq2seq model was used as the base model for the experiments. It incorporates basic attention and word embeddings.

2.3 Beam Search

Beam search was implemented as part of the seq2seq model. It acted as a prediction layer for the decoder portion of the network. Initially it was designed using basic algorithms but was later updated to make effective use of the GPU. To maximize the use of the GPU tensor volumes were designed to hold all necessary information to beam search and beams were combined to make larger batches and decode all beams in parallel. The overall size of batch size then became batch_size * beam_width. This approach made beam search extremely fast but may have created some limitations in the effectiveness of beam search.

2.4 CopyNet

CopyNet was designed with the standard seq2seq decoder as the base decoder. The output from this model was used to generate the attentive read weights and the generation scores. Within this decoder, the selective read weights were updated, multiplied by the endoder outputs, and combined with attentive weights to be used as input to the underlying RNN. Since both the attentive weights and selective weights need to be masked in order to remove unwanted tokens, both are softmaxed immediately after applying the masks. The copy scores are generated by a separate model which is a single-layer neural network with a non-linearity on the output, specifically tanh as was recommended by the authors in (Gu et al., 2016). The output of this network is then multiplied by the hidden states.

Finally, the CopyNet model returns the combined scores for the union of the target and source vocabularies.

3 Experiments

For each model 10 epochs we run to evaluate the output. In this case 10 was sufficient since the models tended to overfit very quickly on this data. We consistently observed a steep increase in loss and perplexity after the seventh epoch. All experiments were run on the multi30k dataset. For beam search, only $k = 3$ and $k = 10$ were reported here since other values generated almost identical results to one of these two. For each experiment attention weights were collected were necessary and the BLEU score was determined.

Interestingly, the base seq2seq outperformed all other models in terms of test perplexity and loss. CopyNet achieved the highest BLEU score and seemed to generalize better when generating translations.

4 Results

4.1 Training & Validation

From 2, we can see that all models besides CopyNet showed very similar curves. CopyNet started with higher numbers and progressively worked towards matching or exceeding the numbers of the other models.

4.2 BLEU

CopyNet achieved a significantly higher BLEU score than both beam search and the base model. The beam search model only performed slightly better than the base.

Model	**BLEU**
CopyNet	**31.80**
Beam Search ($k = 3$)	28.92
`seq2seq`	28.72
Beam Search ($k = 10$)	27.48

Table 1: BLEU Scores for select models.

Model	Test Loss	Test Ppl.
CopyNet	3.283	26.653
Beam Search ($k = 3$)	3.226	25.166
Beam Search ($k = 10$)	3.223	25.093
`seq2seq`	**3.189**	**24.254**

Table 2: Test perplexity (Ppl.) and loss for select models.

4.3 Samples

Sample translations are shown in 3. CopyNet seemed to generate the best translations in all cases. The base model also performed well in translation while beam search didn't perform as well as expected.

4.4 Attention

Below we can see visualization of the attention weights for the attentive read model and the selective read model 3. From the visualizations we can see where the attention model focused more and what positions the copying model chose when copying from source to target. We can see that in every case it used the source sequence as an indicator for when to use the period for the end of a sentence.

5 Conclusion

CopyNet provides a component that is very valuable in machine translation. The ability to predict out-of-vocabulary words and simply copy tokens from the input sentence works very well in practice. This was especially noted in the end of sequences where it seemed to consistently use the input as an indicator for generating the period.

Beam search did not work as well as one would have hoped. This may be due to implementation details that need to be worked out, or it could be that more datasets need to be utilized to determine what the issue may be. In future work more datasets would be used in all experiments and more decoder models explored.

References

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30k: Multilingual english-german image descriptions.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning.

Evan Walsh. 2019. Incorporating a copy mechanism into sequence-to-sequence models.

Origin	Output
source	’ein’, ’schwarzer’, ’hund’, ’und’, ’ein’, ’gefleckter’, ’hund’, ’kämpfen’, ’.’
target	’a’, ’black’, ’dog’, ’and’, ’a’, ’spotted’, ’dog’, ’are’, ’fighting’
CopyNet	’a’, ’black’, ’dog’, ’and’, ’a’, ’spotted’, ’dog’, ’fight’, ’.’
Beam Search ($k = 3$)	’a’, ’black’, ’dog’, ’and’, ’a’, ’spotted’, ’dog’, ’fighting’, ’.’
Beam Search ($k = 10$)	’a’, ’black’, ’dog’, ’and’, ’a’, ’spotted’, ’dog’, ’.’
`seq2seq2`	’a’, ’black’, ’dog’, ’and’, ’a’, ’spotted’, ’dog’, ’fighting’, ’.’
source	’eine’, ’frau’, ’spielt’, ’ein’, ’lied’, ’auf’, ’ihrer’, ’geige’, ’.’
target	’a’, ’female’, ’playing’, ’a’, ’song’, ’on’, ’her’, ’violin’, ’.’
CopyNet	’a’, ’woman’, ’is’, ’playing’, ’a’, ’song’, ’on’, ’her’, ’violin’, ’.’,
Beam Search ($k = 3$)	’a’, ’woman’, ’playing’, ’a’, ’song’, ’on’, ’her’, ’hip’
Beam Search ($k = 10$)	’a’, ’woman’, ’is’, ’playing’, ’a’, ’violin’, ’on’, ’her’, ’violin’, ’.’
`seq2seq2`	’a’, ’woman’, ’playing’, ’a’, ’song’, ’on’, ’her’, ’violin’, ’.’
source	’die’, ’person’, ’im’, ’gestreiften’, ’shirt’, ’klettert’, ’auf’, ’einen’, ’berg’, ’.’
target	’the’, ’person’, ’in’, ’the’, ’striped’, ’shirt’, ’is’, ’mountain’, ’climbing’, ’.’
CopyNet	’the’, ’person’, ’in’, ’the’, ’striped’, ’shirt’, ’is’, ’climbing’, ’a’, ’mountain’, ’.’
Beam Search ($k = 3$)	’the’, ’person’, ’in’, ’the’, ’striped’, ’shirt’, ’is’, ’climbing’, ’a’, ’mountain’, ’.’
Beam Search ($k = 10$)	’the’, ’person’, ’in’, ’a’, ’striped’, ’shirt’, ’climbing’, ’climbing’, ’a’, ’mountain’, ’.’
`seq2seq2`	’the’, ’person’, ’in’, ’the’, ’striped’, ’shirt’, ’is’, ’climbing’, ’a’, ’mountain’, ’.’

Table 3: Sample translations from each model.

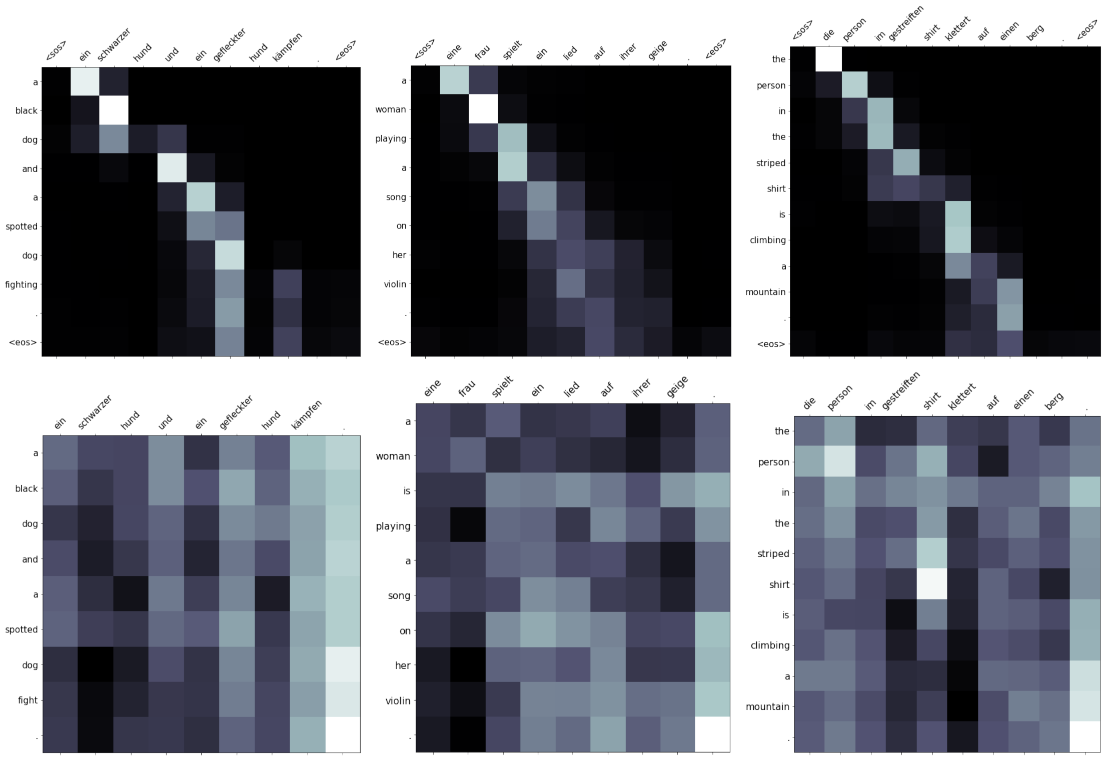

Figure 3: Visualizations of the attentive and selective weights used for the attention mechanisms. The first row is the attentive weights while the second row is the selective weights. Lighter colors mean more attention was placed at these positions of the sequence.