Report of Text Generating

Zhenyang Zhu
zhenyangzhu@buaa.edu.cn

Abstract

This is the forth assignment report for the NLP course. This report utilizes the Seq2Seq model for text generation training, with training data sourced from Jin Yong's novel 《白马啸西风》. Due to hardware limitations, the training results are not entirely satisfactory. However, the model is still able to generate text that somewhat resembles Jin Yong's writing style, showcasing the potential of the Seq2Seq model in text generation.

Methodology

Sequence-to-Sequence (Seq2Seq) models with Long Short-Term Memory (LSTM) have revolutionized the field of natural language processing (NLP) and have found applications in machine translation, text summarization, speech recognition, and more. These models are designed to handle input and output sequences of arbitrary lengths, making them suitable for tasks that involve generating output sequences based on variable-length input sequences. This report will explore the fundamentals of Seq2Seq models with LSTM.

At a high level, Seq2Seq models consist of two recurrent neural networks (RNNs): an encoder and a decoder. The encoder processes the input sequence and compresses the information into a fixed-length context vector or latent representation. This context vector is then used by the decoder to generate the output sequence. LSTMs, a variant of RNNs, are particularly effective in capturing long-term dependencies and handling vanishing or exploding gradient problems, which are common challenges in sequence modeling tasks.

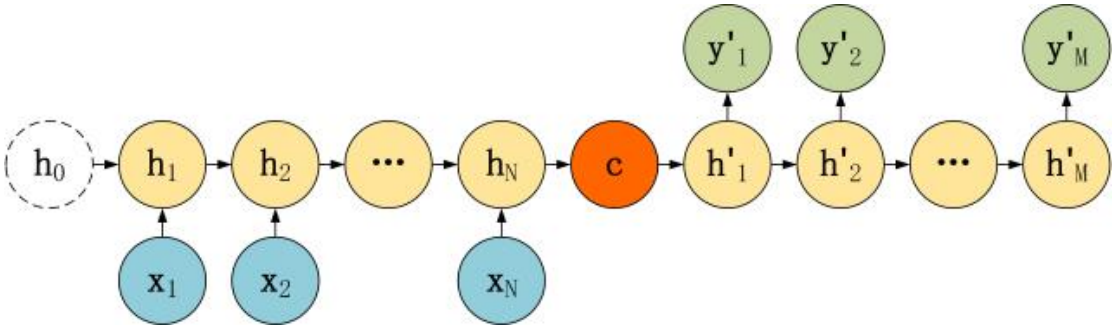

Fig 1 A structure of the Seq2Seq model.

Let's define the mathematical notations used in Seq2Seq models. Suppose we have an input sequence of length T, represented as $X = \{x_1, x_2, ..., x_T\}$, and an output sequence of length U, represented as $Y = \{y_1, y_2, ..., y_U\}$. Each x_i and y_j corresponds to a token in the input and

output sequence, respectively.

The encoder in a Seq2Seq model aims to capture the contextual information of the input sequence and generate a context vector. Let h_t denote the hidden state of the encoder LSTM at time step t. It can be computed using the following equations:

$$h_t = \text{LSTM\_encoder}(x_t, h_{t-1})$$

where LSTM_encoder represents the LSTM cell of the encoder. The final hidden state h_T of the encoder captures the summarized representation of the input sequence.

To generate the output sequence, the decoder LSTM takes the context vector and the previously generated tokens as input. At each time step, the decoder predicts the next token in the output sequence. The hidden state $\tilde{h}_t$ and the output $\tilde{y}_t$ of the decoder LSTM at time step t can be calculated as follows:

$$\tilde{h}_t = \text{LSTM\_decoder}(\tilde{y}_{t-1}, \tilde{h}_{t-1}, c)$$
$$\tilde{y}_t = \text{softmax}(Ws\ \tilde{h}_t + b)$$

where LSTM_decoder represents the LSTM cell of the decoder, c is the context vector obtained from the encoder, Ws is the weight matrix, and b is the bias vector.

The Seq2Seq model is trained to minimize the difference between the predicted output sequence $\tilde{Y} = \{\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_U\}$ and the ground truth output sequence Y. This is typically done by maximizing the log-likelihood of the correct output sequence given the input sequence, which can be formulated as:

$$L = \sum log(P(y_j \mid y_1, ..., y_{j-1}, X))$$

where $P(y_j|y_1, \ldots, y_{j-1}, X)$ represents the probability of generating the j-th token y_j in the output sequence given the previous tokens $y_1, \ldots, y_{j-1}$ and the input sequence X.

During training, the parameters of the Seq2Seq model, including the weights and biases of the encoder and decoder LSTMs, are optimized using techniques like backpropagation through time (BPTT) and gradient descent. This allows the model to learn to generate accurate and meaningful output sequences based on the input sequences.

One of the key advantages of Seq2Seq models with LSTM is their ability to handle variable-length input and output sequences. By using an encoder-decoder architecture, these models can effectively capture the semantic meaning and context of the input sequence and generate coherent and relevant output sequences. This makes them particularly well-suited for tasks such as machine translation, where the input and output sequences can vary significantly in length and structure.

In addition to their flexibility, Seq2Seq models with LSTM also address the issue of vanishing or exploding gradients that often plague traditional RNNs. LSTMs achieve this by introducing a gating mechanism that allows them to selectively remember or forget information over long time intervals. This is crucial for capturing dependencies between distant tokens in a sequence, as it helps prevent the loss of relevant context information.

Experimental Studies

This report utilizes the Seq2Seq model to generate novel texts, using Jin Yong's novel 《白马

啸西风》 as the training data. In terms of text data retrieval, the report first preprocesses the corpus by removing irrelevant words and symbols, and performs tokenization on the text. After tokenization, the text needs to be encoded. The report adopts a simple method by creating a mapping between words and numbers, constructing a dictionary where each word corresponds to a unique index, allowing for bidirectional conversion between words and indices. Subsequently, the training data is prepared. Considering that the input of the Seq2Seq model is a sequence of texts, the report sets the length of each sequence to 50 words and divides the retrieved text into a collection of sequences with this length, serving as the training dataset.

Next, the Seq2Seq model is constructed. The report builds a simple Seq2Seq model, consisting of an encoder and a decoder, both comprised of an Embedding layer and an LSTM layer. Additionally, the decoder is equipped with a fully connected layer for the final output. The specific structure of the constructed Seq2Seq model is illustrated in Figure 2.

```
Seq2Seq(
  (encoder): NumEncoder(
    (embedding): Embedding(448, 256)
    (lstm): LSTM(256, 128, num_layers=2, batch_first=True, dropout=0.1)
  )
  (decoder): NumDecoder(
    (embedding): Embedding(448, 256)
    (lstm): LSTM(256, 128, num_layers=2, batch_first=True, dropout=0.1)
    (fc): Linear(in_features=128, out_features=448, bias=True)
  )
)
```

Fig 2 The parameters of the constructed Seq2Seq model

The processed corpus data is then fed into the model for training and computation. Due to limitations in device performance, this report only trained for 50 epochs, and the variation of loss after each epoch is depicted in Figure 3. The final loss value of the model is 0.4.

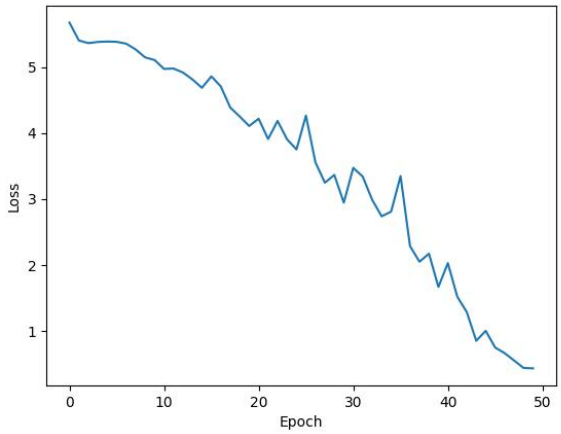

Fig 3 The curve of training loss

Next, a segment from 《白马啸西风》 is selected as input, and the trained Seq2Seq model is used for text generation. Since the report did not set any text start or end identifiers such as "<BOS>" or "<EOS>", the length of the generated text will be the same as the length of the input text. The input text is as follows:

那少妇远远听得丈夫的一声怒吼，当真是心如刀割：「他已死了，我还活著干麼？」从怀中取出一块羊毛织成的手帕，塞在女儿怀里，说道：「秀儿，你好好照料自己！」

挥马鞭在白马臀上一抽，双足一撑，身子已离马鞍。但见那白马鞍上一轻，驮著女孩儿如风疾驰，心中略感安慰：「此马脚力天下无双，秀儿身子又轻，这一下，他们再也追她不上了。」前面，女儿的哭喊声「妈妈，妈妈」渐渐隐去，身後马蹄声却越响越近，

The predicted result of the model is:

背心上却插著一枝长箭。鲜血从他背心流到马背上，又流到地下，滴入了黄沙之中。他不敢伸手拔箭，只怕这枝箭一拔下来，就会支持不住，立时倒毙。谁不死呢？那也，温柔的一笑，说道：「我俩一起经历过无数危难，这次或许也能逃脱。『吕梁三杰』不但要地图，他们……他们还为了你。那少妇道：「他……他总该还有几分情分」在一声，在他右肩刺了进去。拔枪出来，鲜血直喷，白马李三仍是不动。领头的虬髯汉子道：「死得透了，还怕甚麼？快搜他身上

Indeed, it can be observed that the generated text from the model contains some elements of Jin Yong's novels, but the actual content may be incoherent and does not adhere to typical Chinese language usage. This could be attributed to factors such as insufficient training iterations, limited training data, or suboptimal hyperparameter settings. However, even in this rough environment, the Seq2Seq model is still able to generate text that somewhat resembles actual content, showcasing its practical utility.

Conclusion

This report attempts to use the Seq2Seq model for Chinese text generation, aiming to generate a portion of Jin Yong's novel. Due to time and hardware limitations, the final results are not perfect. However, it is still evident that Seq2Seq is a powerful model for text generation. It remains a good choice for building conversational agents and conducting machine translation tasks.