Seq-2-Seq Model for Chatbots (DNLP)

Pengfei (Leonard) Li

Introduction and Motivation

Chatbot is one of the most prevailing and state-of-art deep learning model. Numerous top-tier high-tech and innovative companies investigate a lot into chatbots. A growing number of startups built a lot of bot platforms and bot libraries. One can take a look at the Microsoft's latest product bot developer framework.

There are two ways to realize a chatbot, retrieval-based models and generative models. The former one chooses the response from a bag of responses, i.e. answers. The latter generates the responses word by word. And thanks to sequence modelling in deep learning, it is less likely to produce many grammatical errors. The chatbot below, which belongs to the second type, utilizes NLP and deep learning to mimic the human-like conversations.

Problem definition

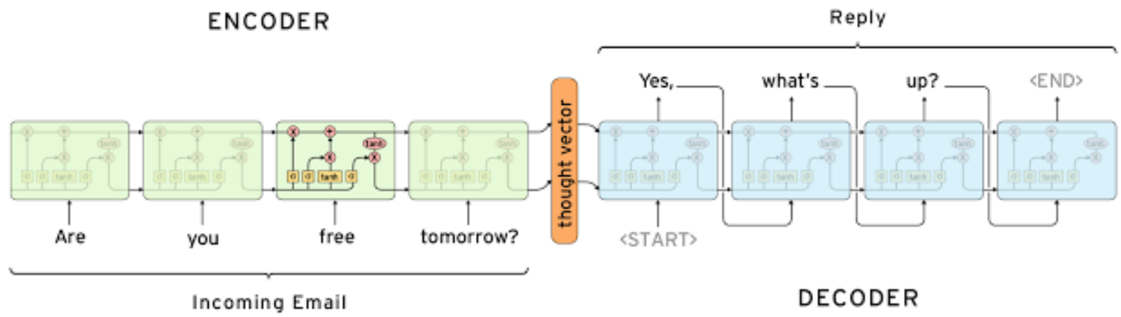
The seq-2-seq model is the whole point of chatbot known as a conversation agent. We propose a query and then the PC provides an answer. As a machine learning, the inputs and outputs are clarified as follows.

- The inputs and outputs are English questions and answers in our everyday life. They are the oral sentences or phrases we use in our daily conversations (or the dialogs in the movie).
- Examples of the input: 'Right. See? You're ready for the quiz.', 'hi.', and 'Sure have.'
- Examples of the output: 'I don't want to see that through.', 'Looks like things worked out tonight.', and 'I really, really wanna go.'

Solution

As mentioned above, our model is a generative model, or specifically, is a seq-2-seq model that consists of two stacked LSTMs. The major concern is that the output should be ordered grammatically. Thus, the LSTM is well-suited to solve sequence modelling. Comparing to RNN, it can save the short-term memory for a long time thus mimic the mechanism of English conversation. It is also good at dealing with gradient exploding and vanishing.

Design



I will discuss and the details upon the seq-2-seq model. The diagram above provides us a big picture of the seq-2-seq model. As we can see, the input is "Are you free tomorrow?" and the output "Yes, What's up?". Each word of the input is fed into each left LSTM cell one by one sequentially. Meanwhile, each word of the output is the return of each right LSTM cell, and every previous word is also fed into the next LSTM cell.

Architecture (Algorithm)

The first part of model is NLP (Natural Language Processing). Details are in the implementation section.

Like normal machine learning, I will split the dataset into training, validation and test. Then the evaluation of the model is a big problem. Common metrics such as BLEU is not suitable. In fact, in How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation researchers find that none of the commonly used metrics really correlate with human judgment. Thus, our model is designed in this way. Training and validation set of 85/15 ratio of the whole dataset. The loss function of training and validation is the cross-entropy function. We justify our result by asking our own questions, which can be seen as the testing dataset.

The deep learning part consists of several structures in order to both improve the accuracy of the model and deal with training issue, since it is a fairly complicated deep learning model.

- The two LSTMs are actually (vertically) stacked LSTMs with multiple layers. They are called encoder RNN and decoder RNN
- Word embeddings is applied to convert the literal words into vectors that LSTM can handle.
- Attention-based mechanism is on the top of decoder RNN can improve the output by considering the source LSTM
- Learning rate decay, Adam optimizer and clipped gradient are used to optimize the training process

- The Dropout method can prevent the seq-2-seq model from overfitting.
- Fully-connected layer finally retrieve the answer.

Data

My data is the Cornell Movie Dialogs Corpus. One can follow the link to dive deep into the dataset. The collectors are Cristian Danescu-Niculescu-Mizil and Lillian Lee. They made the data from IMBD movie scripts database. They handled the data in a systematical way, e.g. discarding the movies with less that 5 votes, removing the pairs with less than 5 conversational changes, etc. It has 220,579 conversational exchanges between 10,292 pairs of movie characters. The sentences are of various lengths.

Implementation

Deeping learning is mainly discussed above. Nevertheless, the seq-2-seq model is actually a DNLP (deep learning and natural language processing) model. Before the deep learning, NLP is applied, which is also significant.

The "re" python library is a great tool to do the NLP. I parsed the original file and extracted the questions the answers in dialogues. Questions and answers are two separate sets. Then I cleaned all the texts by recovering full forms from contractions and removed unnecessary special symbols in sentence. The questions and answers, both with length between 2 and 25, were chosen and sorted. Every word was assigned a unique integer to prepare for word embeddings, in answers set and questions set respectively. The functions I used are "re.sub", and python built-in lists and dictionaries.

The "TensorFlow" python library is employed to build the deep learning model. I assembled the graph based upon the architecture above. Then I executed the computation. The functions used here are "tf.placeholder", "tf.strided_slice", "tf.contrib.rnn.BasicLSTMCell" and etc.

Results

The following shortcuts are my final training and testing results. However, I did not find very powerful GPUs free at the beginning. Thanks to help from my friends, I trained a little bit more, and obtained

```
                          Epoch:  60/100,
                          Batch: 1700/4109,
                          Training Loss Error: 11.652,
                          Training Time on 100 Batches: 27 seconds


                          Epoch:  60/100,
                          Batch: 1800/4109,
                          Training Loss Error: 11.935,
                          Training Time on 100 Batches: 26 seconds


                          Epoch:  60/100,
                          Batch: 1900/4109,
                          Training Loss Error: 11.878,
                          Training Time on 100 Batches: 20 seconds


                          Epoch:  60/100,
                          Batch: 2000/4109,
                          Training Loss Error: 11.824,
                          Training Time on 100 Batches: 27 seconds

Validation Loss Error: 11.360,
                          Batch Validation Time: 48 seconds
Sorry I do not speak better, I need more practice.
My apologies, I cannot speak better anymore.
Game Over
```

```
You: gekj
ChatBot:  I am not sure.

You: hello
ChatBot:  I am not sure I have got aout.

You: get out
ChatBot:  I am sorry I just wanted to see you.

You: do you like movie
ChatBot:  I am not sure I have got aout.

You: how are you
ChatBot:  I am sorry.

You: you are the best
ChatBot:  I am not sure.

You: you mean i would get a chance to talk to her
ChatBot:  I am not sure I do not know.

You: i am serious man he is whacked  he sold his own liver on the black market so he could buy
new speakers
ChatBot:  I am not sure I am not.

You: Goodbye
```

Related Works

It is a very hot topic. And many others have a lot of fancy results.

- Chatbots with Seq2Seq by **Suriyadeepan Ram**, and Pytorch-chatbot by **Alexis David Jacq**
- I also recommend reading Deep Learning for Chatbots, Part 1 – Introduction

Conclusion

Deep learning is a boosting and extremely comprehensive topic. And maybe what most of us know is a tip of iceberg. In this object, the DNLP can provide a brain simulation to let the machine mimic the behavior of a human. And it can improve the accuracy and avoid gradient exploding. However, this project is done lees a month. So, the results are moderately reasonable but with ubiquitous “I am not sure”. I doubt I was stuck somewhere when I performed the gradient descent or there is some inner flaw in the algorithm.

From my perspective, several methods could be conducted in order to improve the chat, finely tuning the hyperparameters and both NLP and deep learning, adjusting the workflow of the seq-2-seq model, etc.