Probabilistic Programming

Erik Gustav Rosvall
Sven Hermans

March 2018

1 Preprocessing text

Before feeding the data to a neural network, the text must be preprocessed. We created a vocabulary that maps words to integers, ordered by frequency of occurrence in the training data. Since this is a one-to-one mapping, it is invertible. Further improvements regarding this vocabulary are possible, for example we could map the multiple of a word to the same integer as its singular, and get rid of basic articles.

As the sentences don't have the same lengths, we pad the sequences with zeros, this way our input layer can expect fixed dimensions. If a test sample would be longer than every training sample, there would be information loss. Finally, the integer representation is converted to a one-hot representation.

2 Autoencoder

The autoencoder consists of an encoder and a decoder, we have chosen to stack two LSTM layers for the encoder to reach a high dimensionality reduction.

- Encoder

  - LSTM layer (1789 ->140)
  - LSTM layer (140 ->50)

- Decoder

  - LSTM layer (50 ->1789)
  - Dense (softmax) layer (1789 ->1789)

When training this neural network for 30 epochs, we reach an accuracy of 98% for the test data, which is very high. For example the first question of the test data:

what is on the left side of the white oven on the floor and on right side of the blue armchair in the image1 ?

becomes:

what is on the left side of the white black on the floor and on right side of the papers pot in the

3   Question answering

For this neural network, we use exactly the same encoder as the previous part. The second part of the network now only consists of one Dense (softmax) layer. When training this neural network for 30 epochs, the categorical accuracy is computed to be 90%. This should be interpreted with a large grain of salt. Since there are multiple answers for some questions, we pad all the sequences with zeros such that every question has 7 answers. The neural network is then correct for most of the empty answers, which explains the high accuracy. We will try to solve this by adding a masking layer that ignores the zeros. The answer of the first question listed above is:

garbage_bin

While the output of our neural network results in:

bottle 2 bottle garbage

This illustrates that the network already learned something, but it still doesn't make sense. For most of the questions, the network just answers 'bottle' or '2'. We will further improve this part of the task.