Machine Learning Engineer Nanodegree

Capstone Proposal

Sanya Saxena

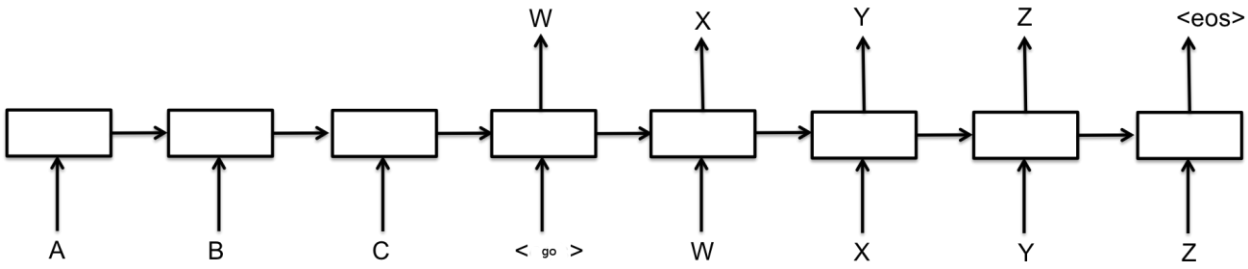
June 20, 2018

Project Overview: German to English Translator

One of the earliest goals for computers was the automatic translation of text from one language to another. Automatic or machine translation is perhaps one of the most challenging artificial intelligence tasks given the fluidity of human language. Classically, rule-based systems were used for this task, which were replaced in the 1990s with statistical methods. More recently, deep neural network models achieve state-of-the-art results in a field that is aptly named neural machine translation.

Domain Background:

A basic sequence-to-sequence model consists of two recurrent neural networks (RNNs): an *encoder* that processes the input and a *decoder* that generates the output. This basic architecture is depicted below.



Each box in the picture above represents a cell of the RNN, most commonly a GRU cell or an LSTM cell. Encoder and decoder can share weights or, as is more common, use a different set of parameters. Multi-layer cells have been successfully used in sequence-to-sequence models too.

Sequence prediction often involves forecasting the next value in a real valued sequence or outputting a class label for an input sequence.

This is often framed as a sequence of one input time step to one output time step (e.g. one-to-one) or multiple input time steps to one output time step (many-to-one) type sequence prediction problem.

Neural machine translation (NMT) is an approach to machine translation that uses a large artificial neural network to predict the likelihood of a sequence of words, typically modeling entire sentences in a single integrated model.

Specifically, an NMT system first reads the source sentence using an *encoder* to build a "thought" vector, a sequence of numbers that represents the sentence meaning; a *decoder*, then, processes the sentence vector to emit a translation. This is often referred to as the *encoder-decoder architecture*. In this manner, NMT addresses the local translation problem in the traditional phrase-based approach: it can capture *long-range dependencies* in languages, e.g., gender agreements; syntax structures; etc., and produce much more fluent translations as demonstrated by Google Neural Machine Translation systems.

Deep learning applications appeared first in speech recognition in the 1990s. The first scientific paper on using neural networks in machine translation appeared in 2014, followed by a lot of advances in the following few years. (Large-vocabulary NMT, application to Image captioning, Subword-NMT, Multilingual NMT, Multi-Source NMT, Character-dec NMT, Zero-Resource NMT, Google, Fully Character-NMT, Zero-Shot NMT in 2017) In 2015 there was the first appearance of a NMT system in a public machine translation competition (OpenMT'15). WMT'15 also for the first time had a NMT contender; the following year it already had 90 % of NMT systems among its winners.

The NMT technique is experiencing considerable development under the fast paced and highly competitive environment. At the latest ACL 2017 conference, all 15 papers accepted under the machine translation category are about the neural machine translation. Therefore, we have every reason to believe that NMT will achieve greater breakthroughs.

Problem Statement:

In this project, we are trying to build a German to English translator with the concept of Neural Machine Translation. The project will take German words as inputs and translate the given words to English.

In general it takes up a small problem of the larger domain of language translation problem. The Neural Machine Translation approach that is taken up in this project is with the help of Keras (Sequence to Sequence Model). This successfully takes any one language as input and translates it desired language. In this project, we deal with Germna to English Translation. If the model

translates the input, we encounter success. We will see the input words, their correct English translation and the English prediction by our model. The model is tested by the BLEU score.

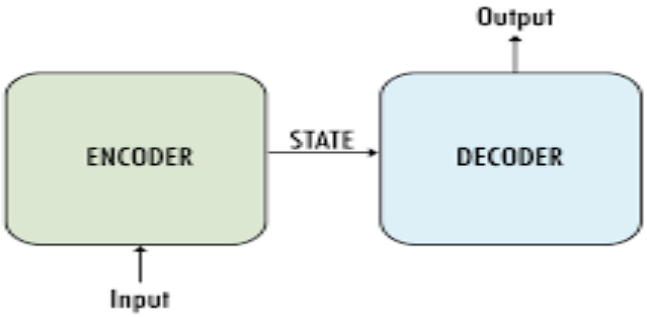
Dataset & Inputs:

We will use a dataset of German to English terms used as the basis for flashcards for language learning.

The dataset is available from the ManyThings.org website, with examples drawn from the Tatoeba Project. The dataset is comprised of German phrases and their English counterparts and is intended to be used with the Anki flashcard software.
The dataset can be downloaded from:
http://www.manythings.org/anki/
The page provides a list of many language pairs.We have a file called *deu.txt* that contains 152,820 pairs of English to German phases, one pair per line with a tab separating the language.
We will frame the prediction problem as given a sequence of words in German as input, translate or predict the sequence of words in English.

Solution Statement:

This project uses Keras as a tool for the machine translation of Geraman to English. We build a sequence to sequence model, where German words are input and English translated words are the output.



We will use an encoder-decoder LSTM model on this problem. In this architecture, the input sequence is encoded by a front-end model called the encoder then decoded word by word by a backend model called the decoder.

The model is trained using the efficient Adam approach to stochastic gradient descent and minimizes the categorical loss function because we have framed the prediction problem as multi-class classification. Evaluation involves two steps: first generating a translated output sequence, and then repeating this process for many input examples and summarizing the skill of the model across multiple cases.

Starting with inference, the model can predict the entire output sequence in a one-shot manner. We will also calculate the BLEU scores to get a quantitative idea of how well the model has performed.

Benchmark Model:

Machine translation has significantly evolved over time, especially in terms of accuracy levels in its output. In the present scenario, one of the well known models of this area is ***Google Translate***. It can translate any language into the desired language.

In the above stated work, we cite GNMT, ***Google's Neural Machine Translation system***. This model consists of a deep LSTM network with 8 encoder and 8 decoder layers using attention and residual connections. To improve parallelism and therefore decrease training time, its attention mechanism connects the bottom layer of the decoder to the top layer of the encoder. To accelerate the final translation speed, it employs low-precision arithmetic during inference computations. To improve handling of rare words, it divides words into a limited set of common sub-word units ("word pieces") for both input and output. This method provides a good balance between the flexibility of "character"-delimited models and the efficiency of "word"-delimited models, naturally handles translation of rare words, and ultimately improves the overall accuracy of the system.

Evaluation Metrics:

The project can be evaluated on basis of BLEU score. **BLEU** (**bilingual evaluation understudy**) is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another. Quality is considered to be the correspondence between a machine's output and that of a human: "the closer a machine translation is to a professional human translation, the better it is" – this is the central idea behind BLEU.

BLEU’s output is always a number between 0 and 1. This value indicates how similar the candidate text is to the reference texts, with values closer to 1 representing more similar texts.

Project Design:

The project aims at translation of German words to English.

Initially we have our dataset, which is a pair of German to English translated words. We clean our data by removing punctuations, lower-upper cases,duplicates etc. We then store the cleaned data in a file. Then the data is loaded and splitted into training and testing sets. We tokenise the data with Keras that map words to integers, differently for English words and German words. Each input and output sequence is encoded to integers and padded to maxium phrase length. Output English sequence is one-hot encoded, as the model will predict the probability of each word in the output. The model is then trained using Adam approach to stochastic gradient descent. LSTM modelling is used. At last model is evaluated using the BLEU score. We hence get English translated words from given German words.

References:

- A Statistical Approach to Machine Translation, 1990.
- Review Article: Example-based Machine Translation, 1999.
- Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, 2014.
- Neural Machine Translation by Jointly Learning to Align and Translate, 2014.
- Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, 2016.
- Sequence to sequence learning with neural networks, 2014.
- Recurrent Continuous Translation Models, 2013.
- Continuous space translation models for phrase-based statistical machine translation, 2013.
- https://github.com/tensorflow/nmt
- https://www.youtube.com/watch?v=nRBnh4qbPHI&t=330s