Sequence to Sequence Modeling

Tony Francis

Problem Formulation

Find a mapping between variable length input and variable length output.

Use Cases

- Machine Translation
- Summarization
- Speech Recognition
- Text to Speech
- Question & Answering
- Image Captioning

Machine Translation

Statistical Translation - Language Model

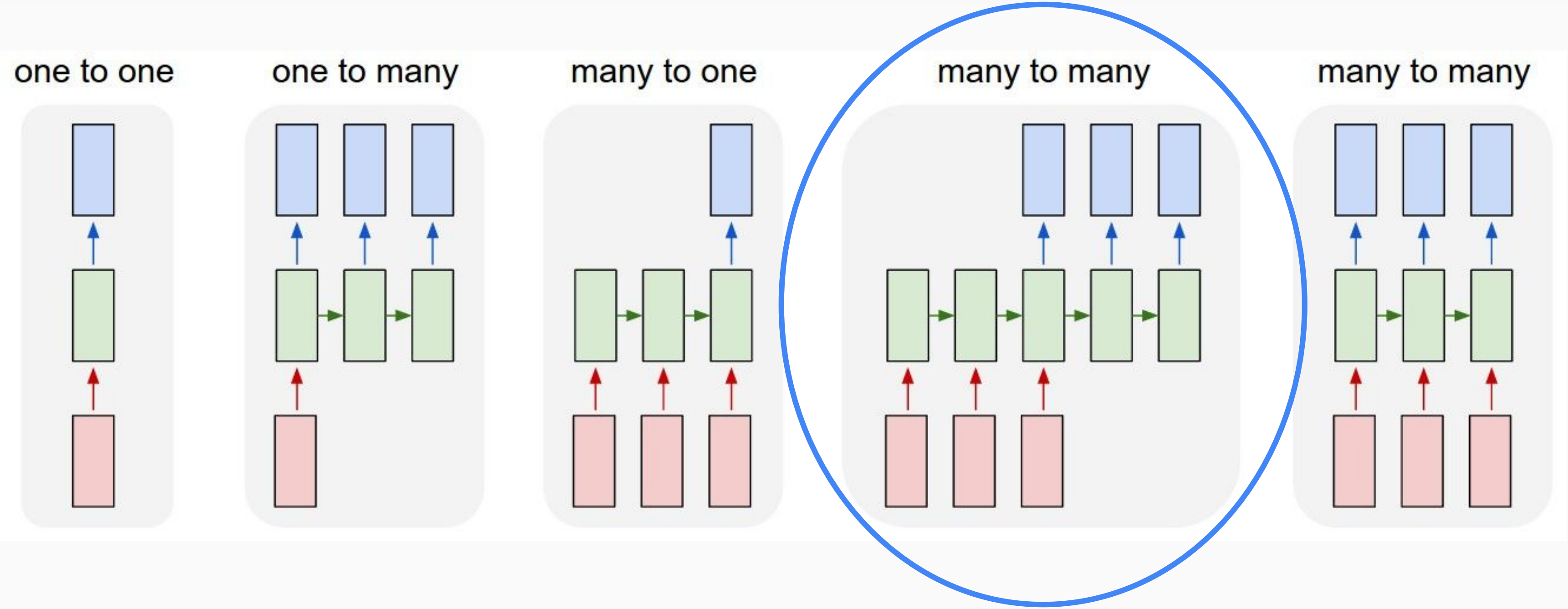
- Goal of a language model: Determine what good English is as P(e)
- Standard Technique: Trigram Model
  - Conditional probability of a word given 2 previous words
  - “Colorless green ideas sleep furiously.”
    - <s> <s> colorless green ideas sleep furiously . </s>
  - colorless => p(colorless | <s> <s>)
  - green => p(green | <s> colorless)
  - ideas => p(ideas | colorless green)
  - sleep => p(sleep | colorless green)
  - furiously => p(furiously | green sleep)
  - . => p(. | sleep furiously)
  - </s> => p(</s> | furiously .)

Statistical Translation - Decoding

- Since we have a one-one mapping of words for a pair of languages. We then employ the language model to “decode” one language into the target language.
- Maximize conditional probabilities

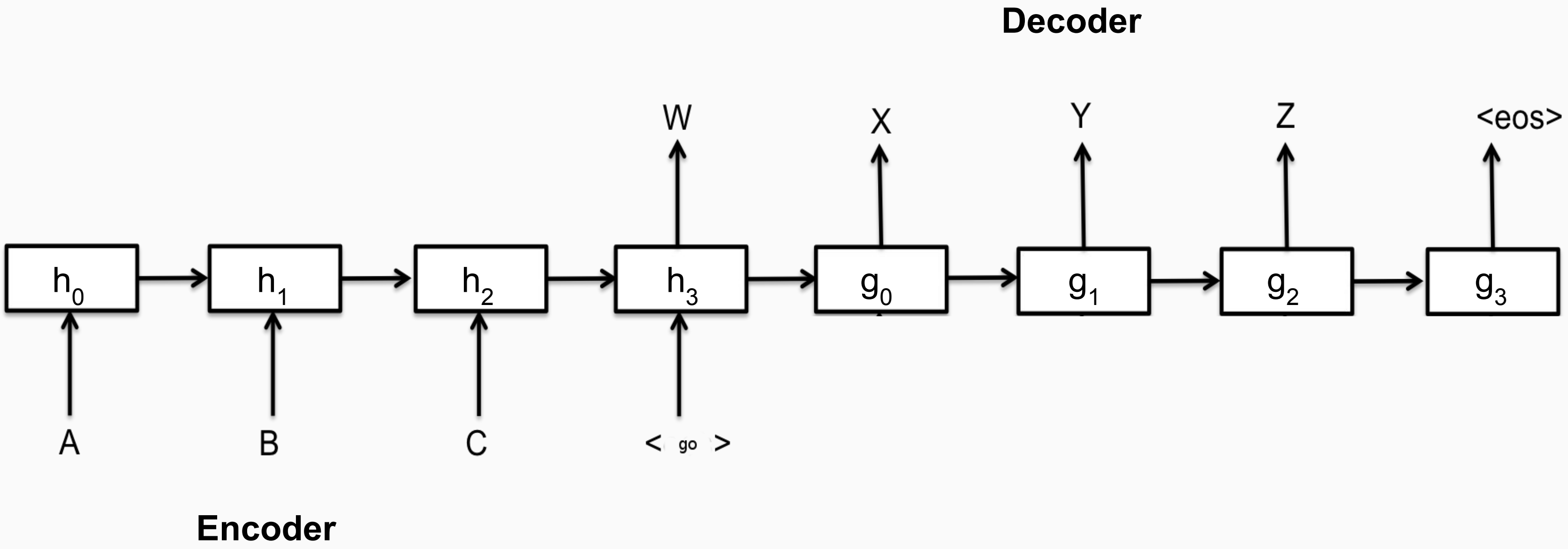
Seq2Seq Approach

RNNs

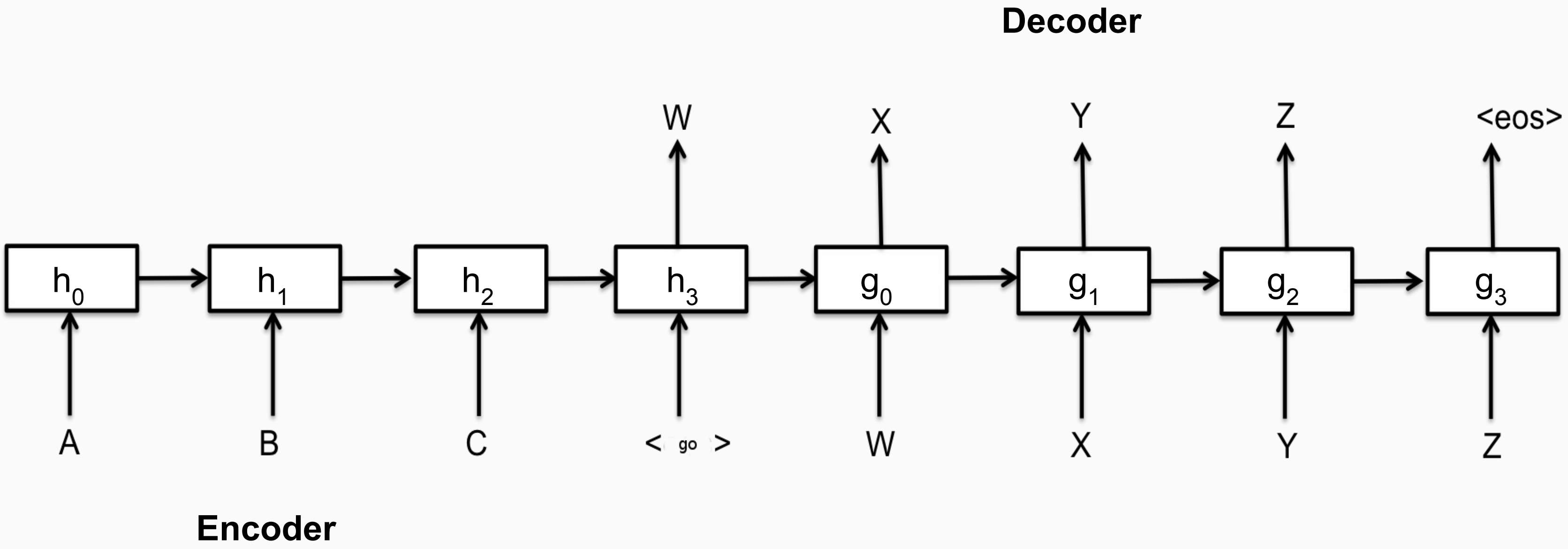


https://karpathy.github.io/2015/05/21/rnn-effectiveness/

Model



Model



Training

- Stochastic Gradient Descent
- Sample an input x, and an output y
  - A sentence in French, and a sentence in English
- Sample a random word y_t in y
- Update RNN Encoder and Decoder parameters to increase probability of y_t given y_{t-1} , y_{t-2} , … , y_0 , x_n , … , x_0

Prediction Algorithms - Greedy Decoding

For an input sequence x:

Given x, find word y_0 with the highest probability

Given y_0 and x, find y_1 with the highest probability

And so on, until reaching y_n = </s>

Prediction Algorithms - Beam Search Decoding

For any input sequence x

Given x, find k candidates for y0 with the highest probabilities

Given x, for each y0 candidate, find k candidates for word y1

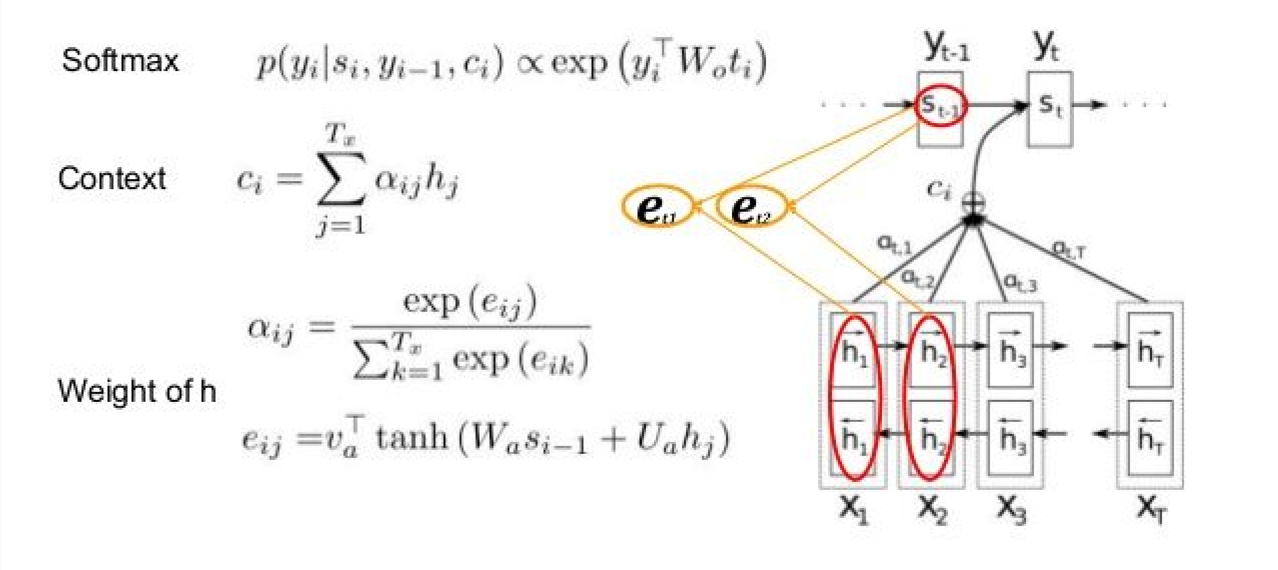
Stop when <END> is produced

Output => the sequence with the highest conditional probability

Attention Mechanism

- Seq2Seq models struggle with longer sentences
- Encoder compresses input series into a representation of fixed size
- Attention Mechanism is placed between Encoder and Decoder
  - Predicts the output y_t with a weighted average context vector, not just the last state
  - This is a parameter that is trained

Attention Mechanism



Softmax $p(y_i|s_i, y_{i-1}, c_i) \propto \exp\left(y_i^\top W_o t_i\right)$

Context $c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$

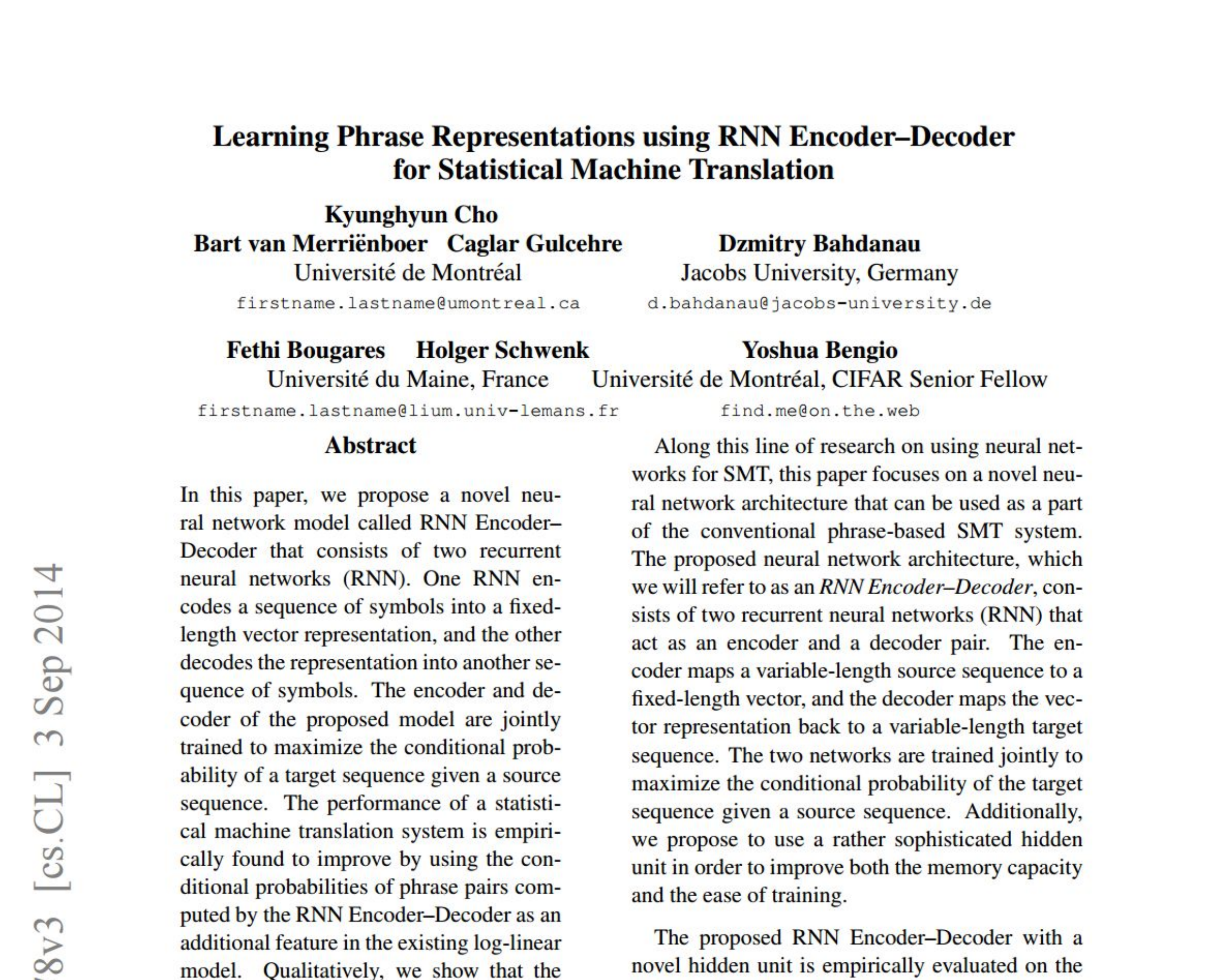
Weight of h

$$\alpha_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(e_{ik}\right)}$$

$$e_{ij} = v_a^\top \tanh\left(W_a s_{i-1} + U_a h_j\right)$$

http://www.slideshare.net/KeonKim/attention-mechanisms-with-tensorflow

Papers to Read



Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

Kyunghyun Cho
Bart van Merriënboer Caglar Gulcehre
Université de Montréal
firstname.lastname@umontreal.ca

Dzmitry Bahdanau
Jacobs University, Germany
d.bahdanau@jacobs-university.de

Fethi Bougares Holger Schwenk
Université du Maine, France
firstname.lastname@lium.univ-lemans.fr

Yoshua Bengio
Université de Montréal, CIFAR Senior Fellow
find.me@on.the.web

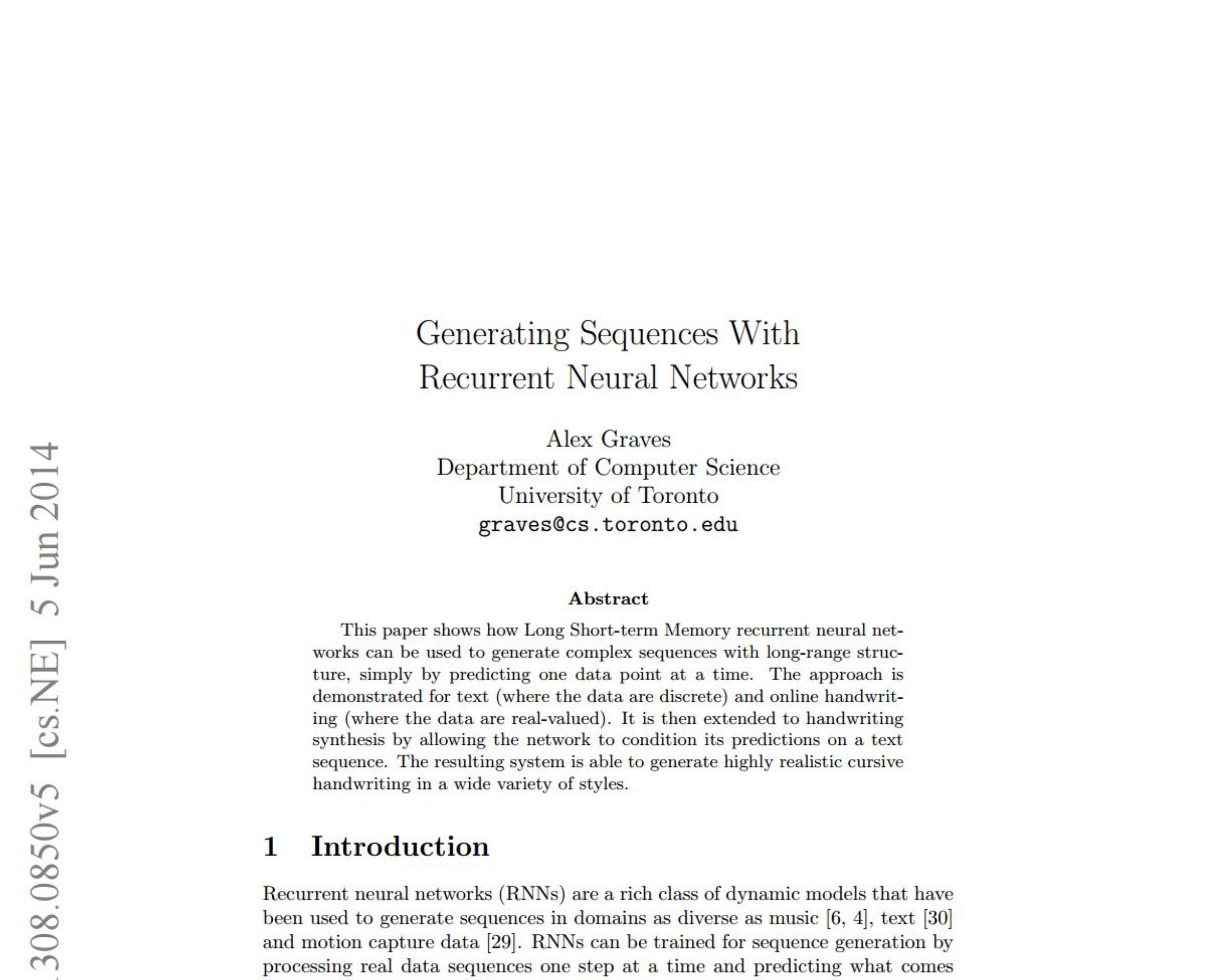
Abstract

In this paper, we propose a novel neural network model called RNN Encoder–Decoder that consists of two recurrent neural networks (RNN). One RNN encodes a sequence of symbols into a fixed-length vector representation, and the other decodes the representation into another sequence of symbols. The encoder and decoder of the proposed model are jointly trained to maximize the conditional probability of a target sequence given a source sequence. The performance of a statistical machine translation system is empirically found to improve by using the conditional probabilities of phrase pairs computed by the RNN Encoder–Decoder as an additional feature in the existing log-linear model. Qualitatively, we show that the

Along this line of research on using neural networks for SMT, this paper focuses on a novel neural network architecture that can be used as a part of the conventional phrase-based SMT system. The proposed neural network architecture, which we will refer to as an *RNN Encoder–Decoder*, consists of two recurrent neural networks (RNN) that act as an encoder and a decoder pair. The encoder maps a variable-length source sequence to a fixed-length vector, and the decoder maps the vector representation back to a variable-length target sequence. The two networks are trained jointly to maximize the conditional probability of the target sequence given a source sequence. Additionally, we propose to use a rather sophisticated hidden unit in order to improve both the memory capacity and the ease of training.

The proposed RNN Encoder–Decoder with a novel hidden unit is empirically evaluated on the

arxiv.org/abs/1406.1078



Generating Sequences With Recurrent Neural Networks

Alex Graves
Department of Computer Science
University of Toronto
graves@cs.toronto.edu

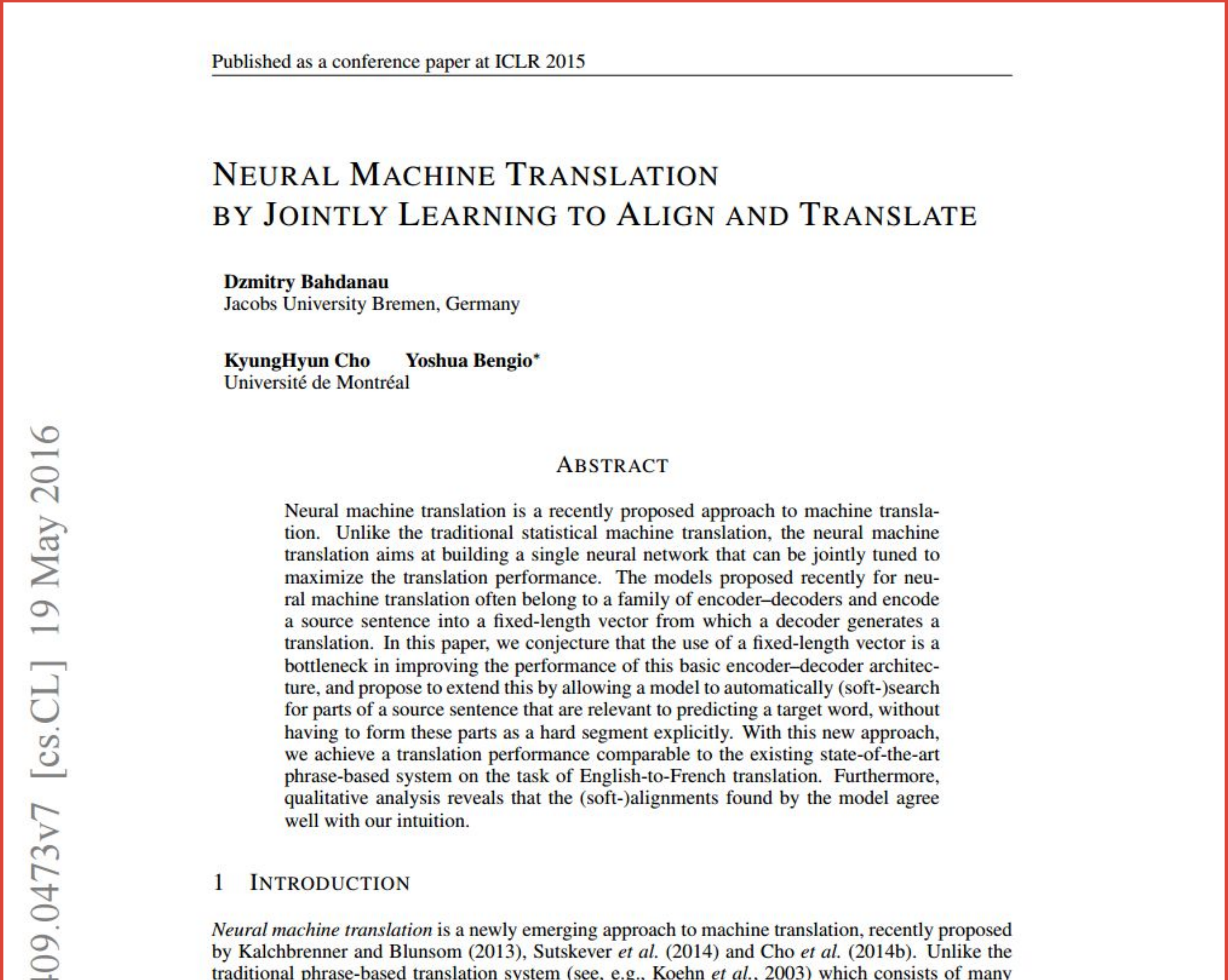
Abstract

This paper shows how Long Short-term Memory recurrent neural networks can be used to generate complex sequences with long-range structure, simply by predicting one data point at a time. The approach is demonstrated for text (where the data are discrete) and online handwriting (where the data are real-valued). It is then extended to handwriting synthesis by allowing the network to condition its predictions on a text sequence. The resulting system is able to generate highly realistic cursive handwriting in a wide variety of styles.

1 Introduction

Recurrent neural networks (RNNs) are a rich class of dynamic models that have been used to generate sequences in domains as diverse as music [6, 4], text [30] and motion capture data [29]. RNNs can be trained for sequence generation by processing real data sequences one step at a time and predicting what comes

arxiv.org/abs/1308.0850



Published as a conference paper at ICLR 2015

NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio***
Université de Montréal

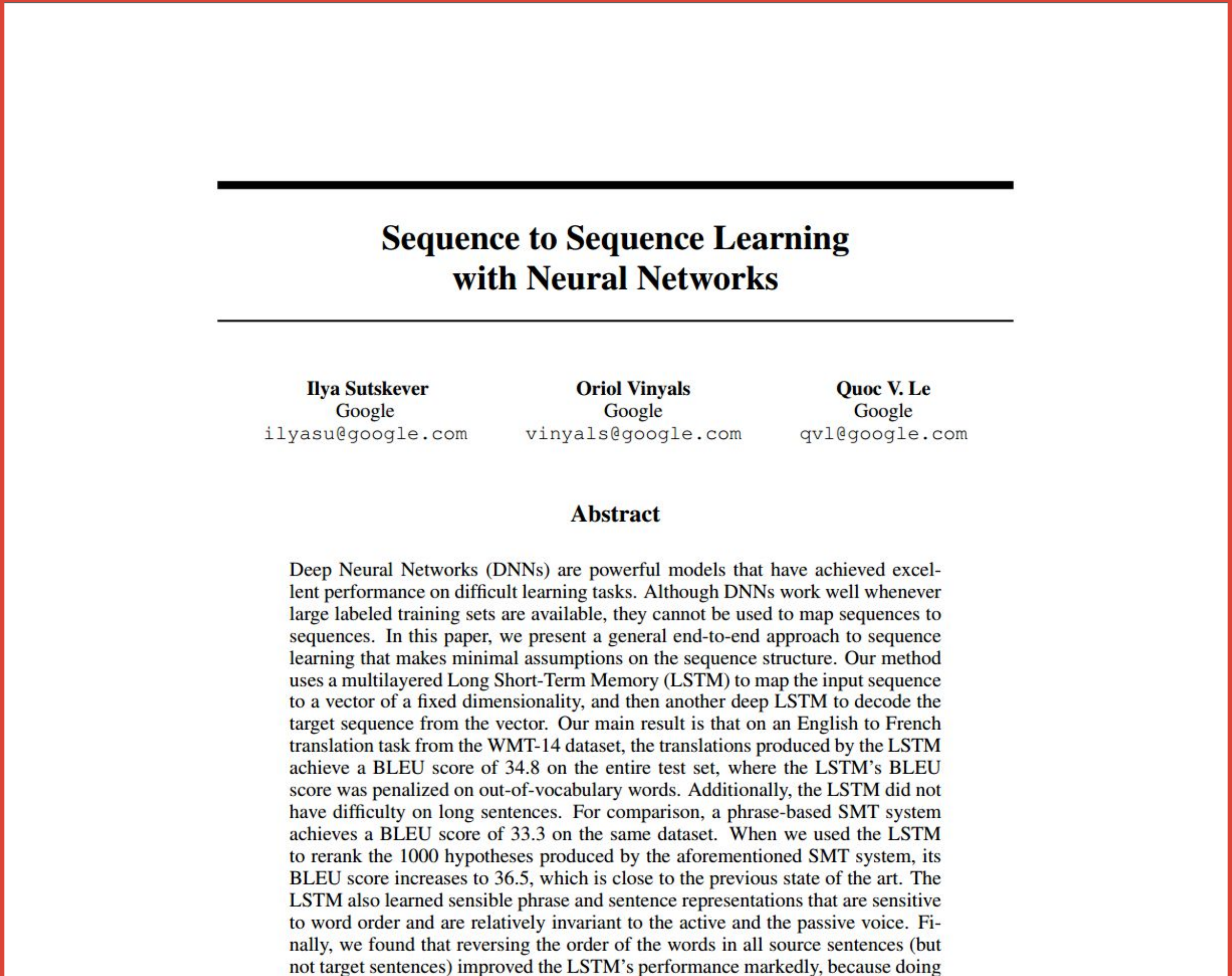
ABSTRACT

Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently for neural machine translation often belong to a family of encoder–decoders and encode a source sentence into a fixed-length vector from which a decoder generates a translation. In this paper, we conjecture that the use of a fixed-length vector is a bottleneck in improving the performance of this basic encoder–decoder architecture, and propose to extend this by allowing a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without having to form these parts as a hard segment explicitly. With this new approach, we achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. Furthermore, qualitative analysis reveals that the (soft-)alignments found by the model agree well with our intuition.

1 INTRODUCTION

Neural machine translation is a newly emerging approach to machine translation, recently proposed by Kalchbrenner and Blunsom (2013), Sutskever *et al.* (2014) and Cho *et al.* (2014b). Unlike the traditional phrase-based translation system (see, e.g., Koehn *et al.*, 2003) which consists of many

arxiv.org/abs/1409.0473



Sequence to Sequence Learning with Neural Networks

Ilya Sutskever
Google
ilyasu@google.com

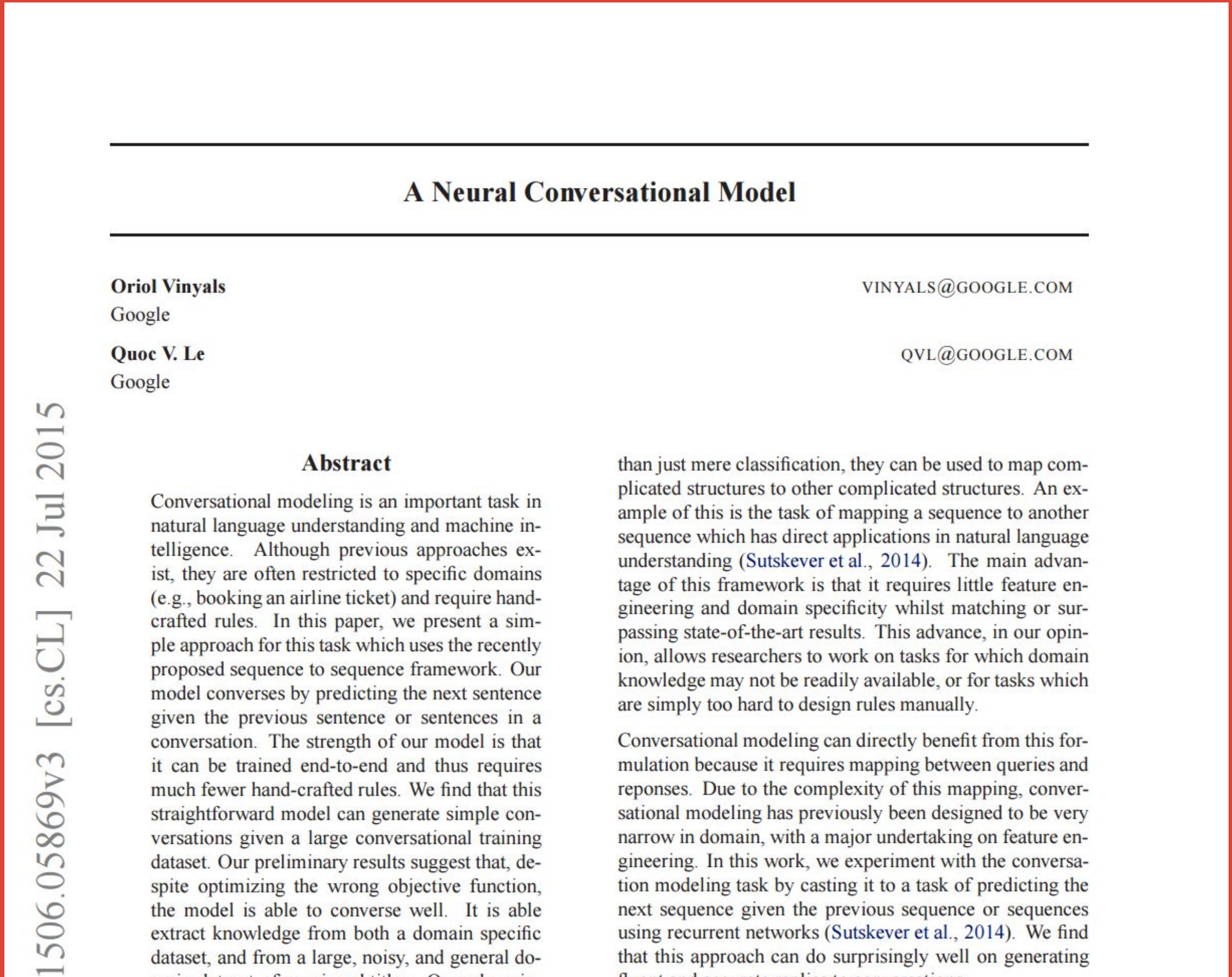
Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

Abstract

Deep Neural Networks (DNNs) are powerful models that have achieved excellent performance on difficult learning tasks. Although DNNs work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. In this paper, we present a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimensionality, and then another deep LSTM to decode the target sequence from the vector. Our main result is that on an English to French translation task from the WMT-14 dataset, the translations produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where the LSTM's BLEU score was penalized on out-of-vocabulary words. Additionally, the LSTM did not have difficulty on long sentences. For comparison, a phrase-based SMT system achieves a BLEU score of 33.3 on the same dataset. When we used the LSTM to rerank the 1000 hypotheses produced by the aforementioned SMT system, its BLEU score increases to 36.5, which is close to the previous state of the art. The LSTM also learned sensible phrase and sentence representations that are sensitive to word order and are relatively invariant to the active and the passive voice. Finally, we found that reversing the order of the words in all source sentences (but not target sentences) improved the LSTM's performance markedly, because doing

https://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf



A Neural Conversational Model

Oriol Vinyals VINYALS@GOOGLE.COM
Google

Quoc V. Le QVL@GOOGLE.COM
Google

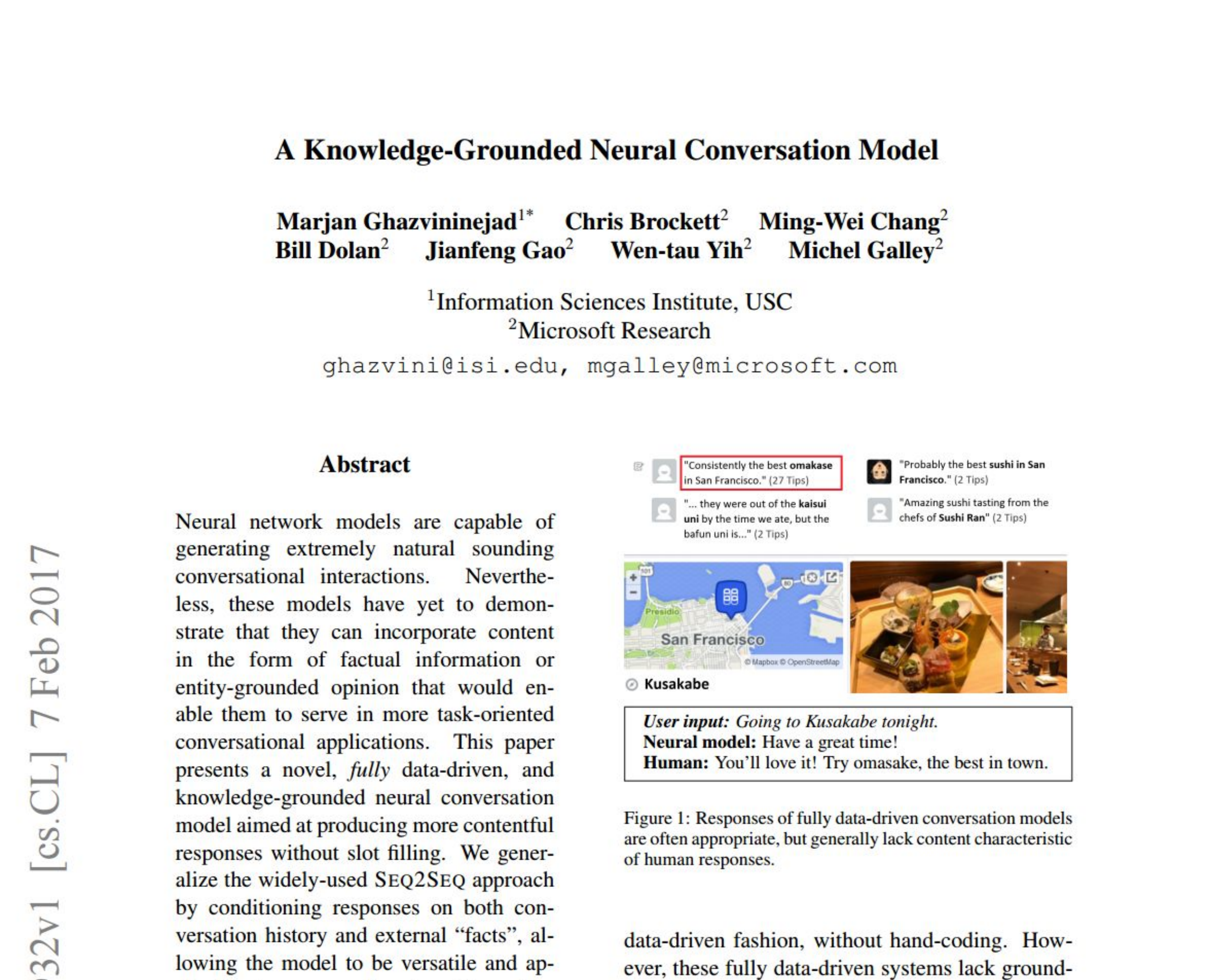
Abstract

Conversational modeling is an important task in natural language understanding and machine intelligence. Although previous approaches exist, they are often restricted to specific domains (e.g., booking an airline ticket) and require hand-crafted rules. In this paper, we present a simple approach for this task which uses the recently proposed sequence to sequence framework. Our model converses by predicting the next sentence given the previous sentence or sentences in a conversation. The strength of our model is that it can be trained end-to-end and thus requires much fewer hand-crafted rules. We find that this straightforward model can generate simple conversations given a large conversational training dataset. Our preliminary results suggest that, despite optimizing the wrong objective function, the model is able to converse well. It is able extract knowledge from both a domain specific dataset, and from a large, noisy, and general do-

than just mere classification, they can be used to map complicated structures to other complicated structures. An example of this is the task of mapping a sequence to another sequence which has direct applications in natural language understanding (Sutskever et al., 2014). The main advantage of this framework is that it requires little feature engineering and domain specificity whilst matching or surpassing state-of-the-art results. This advance, in our opinion, allows researchers to work on tasks for which domain knowledge may not be readily available, or for tasks which are simply too hard to design rules manually.

Conversational modeling can directly benefit from this formulation because it requires mapping between queries and reponses. Due to the complexity of this mapping, conversational modeling has previously been designed to be very narrow in domain, with a major undertaking on feature engineering. In this work, we experiment with the conversation modeling task by casting it to a task of predicting the next sequence given the previous sequence or sequences using recurrent networks (Sutskever et al., 2014). We find that this approach can do surprisingly well on generating

arxiv.org/abs/1506.05869



A Knowledge-Grounded Neural Conversation Model

Marjan Ghazvininejad[1*] **Chris Brockett**[2] **Ming-Wei Chang**[2]
Bill Dolan[2] **Jianfeng Gao**[2] **Wen-tau Yih**[2] **Michel Galley**[2]

[1]Information Sciences Institute, USC
[2]Microsoft Research

ghazvini@isi.edu, mgalley@microsoft.com

Abstract

Neural network models are capable of generating extremely natural sounding conversational interactions. Nevertheless, these models have yet to demonstrate that they can incorporate content in the form of factual information or entity-grounded opinion that would enable them to serve in more task-oriented conversational applications. This paper presents a novel, *fully* data-driven, and knowledge-grounded neural conversation model aimed at producing more contentful responses without slot filling. We generalize the widely-used SEQ2SEQ approach by conditioning responses on both conversation history and external "facts", allowing the model to be versatile and ap-

"Consistently the best **omakase** in San Francisco." (27 Tips)
"Probably the best **sushi in San Francisco.**" (2 Tips)
"... they were out of the **kaisui uni** by the time we ate, but the bafun uni is..." (2 Tips)
"Amazing sushi tasting from the chefs of **Sushi Ran**" (2 Tips)
Presidio
San Francisco
© Mapbox © OpenStreetMap
Kusakabe

User input: *Going to Kusakabe tonight.*
Neural model: Have a great time!
Human: You'll love it! Try omakase, the best in town.

Figure 1: Responses of fully data-driven conversation models are often appropriate, but generally lack content characteristic of human responses.

data-driven fashion, without hand-coding. However, these fully data-driven systems lack ground-

arxiv.org/abs/1702.01932