Sequence to Sequence Learning for Event Prediction

Dai Quoc Nguyen[1], **Dat Quoc Nguyen**[2], **Cuong Xuan Chu**[3],
Stefan Thater[1], **Manfred Pinkal**[1]

[1]Department of Computational Linguistics, Saarland University, Germany
{daiquocn, stth, pinkal}@coli.uni-saarland.de
[2]Department of Computing, Macquarie University, Australia
dat.nguyen@mq.edu.au
[3]Max Planck Institute for Informatics, Germany
cxchu@mpi-inf.mpg.de

Abstract

This paper presents an approach to the task of predicting an event description from a preceding sentence in a text. Our approach explores sequence-to-sequence learning using a bidirectional multi-layer recurrent neural network. Our approach substantially outperforms previous work in terms of the BLEU score on two datasets derived from WIKIHOW and DESCRIPT respectively. Since the BLEU score is not easy to interpret as a measure of event prediction, we complement our study with a second evaluation that exploits the rich linguistic annotation of gold paraphrase sets of events.

1 Introduction

We consider a task of event prediction which aims to generate sentences describing a predicted event from the preceding sentence in a text. The following example presents an instruction in terms of a sequence of contiguous event descriptions for the activity of baking a cake:

> Gather ingredients. Turn on oven. Combine ingredients into a bowl. Pour batter in pan. Put pan in oven. Bake for specified time.

The task is to predict event description *"Put pan in oven"* from sentence *"Pour batter in pan"*, or how to generate the continuation of the story, i.e., the event following *"Bake for specified time"*, which might be *"Remove pan from oven"*. Event prediction models an important facet of semantic expectation, and thus will contribute to text understanding as well as text generation. We propose to employ sequence-to-sequence learning (SEQ2SEQ) for this task.

SEQ2SEQ have received significant research attention, especially in machine translation (Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015; Luong et al., 2015), and in other NLP tasks such as parsing (Vinyals et al., 2015; Dong and Lapata, 2016), text summarization (Nallapati et al., 2016) and multi-task learning (Luong et al., 2016). In general, SEQ2SEQ uses an *encoder* which typically is a recurrent neural network (RNN) (Elman, 1990) to encode a source sequence, and then uses another RNN which we call *decoder* to decode a target sequence. The goal of SEQ2SEQ is to estimate the conditional probability of generating the target sequence given the encoding of the source sequence. These characteristics of SEQ2SEQ allow us to approach the event prediction task. SEQ2SEQ has been applied to text prediction by Kiros et al. (2015) and Pichotta and Mooney (2016). We also use SEQ2SEQ for prediction of what comes next in a text. However, there are several key differences.

- We collect a new dataset based on the largest available resource of instructional texts, i.e., WIKIHOW[1], consisting of pairs of adjacent sentences, which typically describe contiguous members of an event chain characterizing a complex activity. We also present another dataset based on the DESCRIPT corpus—a crowdsourced corpus of event sequence descriptions (Wanzare et al., 2016). While the WIKIHOW-based dataset helps to evaluate the models in an open-domain setting, the DESCRIPT-based dataset is used to evaluate the models in a closed-domain setting.

[1]www.wikihow.com

- Pichotta and Mooney (2016) use the BLEU score (Papineni et al., 2002) for evaluation (i.e., the standard evaluation metric used in machine translation), which measures surface similarity between predicted and actual sentences. We complement this evaluation by measuring prediction accuracy on the semantic level. To this purpose, we use the gold paraphrase sets of event descriptions in the DESCRIPT corpus, e.g., *"Remove cake"*, *"Remove from oven"* and *"Take the cake out of oven"* belong to the same gold paraphrase set of taking out oven. The gold paraphrase sets allow us to access the correctness of the prediction which could not be attained by using the BLEU measure.

- We explore multi-layer RNNs which have currently shown the advantage over single/shallow RNNs (Sutskever et al., 2014; Vinyals et al., 2015; Luong et al., 2015). We use a bidirectional RNN architecture for the encoder and examine the RNN decoder with or without *attention mechanism*. We achieve better results than previous work in terms of BLEU score.

2 Sequence to Sequence Learning

Given a source sequence $x_1, x_2, ..., x_m$ and a target sequence $y_1, y_2, ..., y_n$, sequence to sequence learning (SEQ2SEQ) is to estimate the conditional probability $\Pr(y_1, y_2, ..., y_n \mid x_1, x_2, ..., x_m)$ (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2015; Vinyals et al., 2015; Luong et al., 2016). Typically, SEQ2SEQ consists of a RNN encoder and a RNN decoder. The RNN encoder maps the source sequence into a vector representation $\boldsymbol{c}$ which is then fed as input to the *decoder* for generating the target sequence.

We use a bidirectional RNN (BiRNN) architecture (Schuster and Paliwal, 1997) for mapping the source sequence $x_1, x_2, ..., x_m$ into the list of encoder states $\boldsymbol{s}^e_1, \boldsymbol{s}^e_2, ..., \boldsymbol{s}^e_m$.

The RNN decoder is able to work with or without *attention mechanism*. When *not* using attention mechanism (Sutskever et al., 2014; Cho et al., 2014), the vector representation $\boldsymbol{c}$ is the last state $\boldsymbol{s}^e_m$ of the encoder, which is used to initialize the decoder. Then, at the timestep i $(1 \leq i \leq n)$, the RNN decoder takes into account the hidden state $\boldsymbol{s}^d_{i-1}$ and the previous input y_{i-1} to output the hidden state $\boldsymbol{s}^d_i$ and generate the target y_i.

Attention mechanism allows the decoder to attend to different parts of the source sequence at one position of a timestep of generating the target sequence (Bahdanau et al., 2015; Luong et al., 2015; Vinyals et al., 2015). We adapt the attention mechanism proposed by Vinyals et al. (2015) to employ a concatenation of the hidden state $\boldsymbol{s}^d_i$ and the vector representation $\boldsymbol{c}$ to make predictions at the timestep i.

We use two advanced variants of RNNs that replace the cells of RNNs with the Long Sort Term Memory (LSTM) cells (Hochreiter and Schmidhuber, 1997) and the Gated Recurrent Unit (GRU) cells (Cho et al., 2014). We also use a deeper architecture of multi-layers, to model complex interactions in the context. This is different from Kiros et al. (2015) and Pichotta and Mooney (2016) where they only use a single layer. So we in fact experiment with Bidirectional-LSTM multi-layer RNN (BiLSTM) and Bidirectional-GRU multi-layer RNN (BiGRU).

3 Experiments

3.1 Datasets

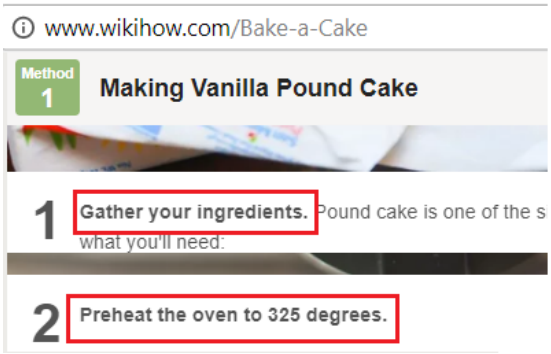

Figure 1: An WIKIHOW activity example.

WIKIHOW-based dataset: WIKIHOW is the largest collection of "how-to" tasks, created by an online community, where each task is represented by sub-tasks with detailed descriptions and pictorial illustrations, e.g., as shown in Figure 1. We collected 168K articles (e.g., "*Bake-a-Cake*") consisting of 238K tasks (e.g., "*Making Vanilla Pound Cake*") and approximately 1.59 millions sub-tasks (e.g., "*Gather your ingredients*", "*Preheat the oven to 325 degrees*"), representing a wide variety of activities and events. Then we created a corpus of approximately 1.34 million pairs of subsequent sub-tasks (i.e., source and target

sentences for the SEQ2SEQ model), for which we have the training set of approximately 1.28 million pairs, the development and test sets of 26,800 pairs in each. This dataset aims to evaluate the models in an open-domain setting where the predictions can go into many kinds of directions.

DESCRIPT-based dataset: The DESCRIPT corpus (Wanzare et al., 2016) is a crowdsourced corpus of event sequence descriptions on 40 different scenarios with approximately 100 event sequence descriptions per scenario. In addition, the corpus includes the gold paraphrase sets of event descriptions. From the DESCRIPT corpus, we create a new corpus of 29,150 sentence pairs of an event and its next contiguous event. Then, for each 10 sentence pairs, the 5th and 10th pairs are used for the development and test sets respectively, and 8 remaining pairs are used for the training set. Thus, each of the development and test sets has 2,915 pairs, and the training set has 23,320 pairs. This dataset helps to assess the models in a closed-domain setting where the goal is trying to achieve a reasonable accuracy.

3.2 Implementation details

The models are implemented in TensorFlow (Abadi et al., 2016) and trained with/without attention mechanism using the training sets. Then, given a source sentence describing an event as input, the trained models are used to generate a sentence describing a predicted event. We use the BLEU metric (Papineni et al., 2002) to evaluate the generated sentences against the target sentences corresponding to the source sentences. A SEQ2SEQ architecture using a single layer adapted by Pichotta and Mooney (2016) is treated as the BASELINE model.

We found vocabulary sizes of 30,000 and 5,000 most frequent words as optimal for the WIKIHOW and DESCRIPT-based datasets, respectively. Words not occurring in the vocabulary are mapped to a special token UNK. Word embeddings are initialized using the pre-trained 300-dimensional word embeddings provided by Word2Vec (Mikolov et al., 2013) and then updated during training. We use two settings of a single BiLSTM/BiGRU layer (1-LAYER-BISEQ2SEQ) and two BiLSTM/BiGRU layers (2-LAYER-BISEQ2SEQ). We use 300 hidden units for both encoder and decoder. Dropout (Srivastava et al., 2014) is applied with probability of 0.5. The training objective is to minimize the cross-entropy loss using the Adam optimizer (Kingma and Ba, 2015) and a mini-batch size of 64. The initial learning rate for Adam is selected from $\{0.0001, 0.0005, 0.001, 0.005, 0.01\}$. We run up to 100 training epochs, and we monitor the BLEU score after each training epoch and select the best model which produces highest BLEU score on the development set.

3.3 Evaluation using BLEU score

Table 1 presents our BLEU scores with models trained on WIKIHOW and DESCRIPT-based data on the respective test sets. There are significant differences in attending to the WIKIHOW sentences and the DESCRIPT sentences. The BLEU scores between the two datasets cannot be compared because of the much larger degree of variation in WIKIHOW. The scores reported in Pichotta and Mooney (2016) on WIKIPEDIA are not comparable to our scores for the same reason.

Model	WIKIHOW		DESCRIPT	
	GRU	LSTM	GRU	LSTM
BASELINE$_{\text{NON-ATT}}$	1.67	1.68	4.31	4.69
1-LAYER-BISEQ2SEQ$_{\text{NON-ATT}}$	2.21	2.01	4.85	5.15
2-LAYER-BISEQ2SEQ$_{\text{NON-ATT}}$	2.53	2.69	4.98	**5.42**
BASELINE$_{\text{ATT}}$	1.86	2.03	4.03	4.01
1-LAYER-BISEQ2SEQ$_{\text{ATT}}$	2.53	2.58	4.38	4.47
2-LAYER-BISEQ2SEQ$_{\text{ATT}}$	**2.86**	2.81	4.76	5.29

Table 1: The BLEU scores on the DESCRIPT and WIKIHOW-based test sets. We use subscripts ATT and NON-ATT to denote models with and without using attention mechanism, respectively.

Table 1 shows that 1-LAYER-BISEQ2SEQ obtains better results than the strong BASELINE. Specifically, 1-LAYER-BISEQ2SEQ improves the baselines with 0.3+ BLEU in both cases of ATT and NON-ATT, indicating the usefulness of using bidirectional architecture. Furthermore, the two-layer architecture produces better scores than the single layer architecture. Using more layers can help to capture prominent linguistic features, that is probably the reason why deeper layers empirically work better.

As shown in Table 1, the GRU-based models obtains similar results to the LSTM-based models on the WIKIHOW-based dataset, but achieves lower scores on the DESCRIPT-based dataset. This could show that LSTM cells with memory gate may help to better remember linguistic fea-

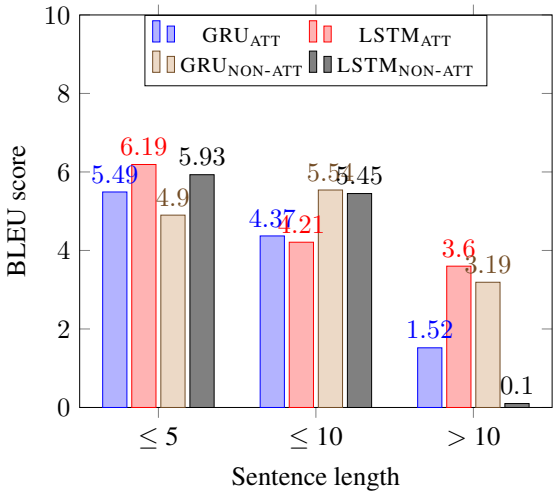

Figure 2: The BLEU scores of two-layer BiLSTM BISEQ2SEQ with/without attention on the DESCRIPT-based test set with respect to the source sentence lengths.

tures than GRU cells without memory gate for the closed-domain setting.

The ATT model outperforms the NON-ATT model on the WIKIHOW-based dataset, but not on the DESCRIPT-based dataset. This is probably because neighboring WIKIHOW sentences (i.e., subtask headers) are more parallel in structure (see *"Pour batter in pan"* and *"Put pan in oven"* from the initial example), which could be related to the fact that they are in general shorter. Figure 2 shows that the ATT model actually works well for DESCRIPT pairs with a short source sentence, while its performance decreases with longer sentences.

3.4 Evaluation based on paraphrase sets

BLEU scores are difficult to interpret for the task: BLEU is a surface-based measure as mentioned in (Qin and Specia, 2015), while event prediction is essentially a semantic task. Table 2 shows output examples of the two-layer BILSTM SEQ2SEQ NON-ATT on the DESCRIPT-based dataset. Although the *target* and *predicted* sentences have different surface forms, they are perfect paraphrases of the same type of event.

To assess the semantic success of the prediction model, we use the gold paraphrase sets of event descriptions provided by the DESCRIPT corpus for 10 of its scenarios. We consider a subset of 682 pairs, for which gold paraphrase information is available, and check, whether a *target* event and its corresponding *predicted* event are paraphrases,

Source:	combine and mix all the ingredients as the recipe delegates
Target:	pour ingredients into a cake pan
Predicted:	put batter into baking pan
Source:	put cake into oven
Target:	wait for cake to bake
Predicted:	bake for specified time
Source:	make an appointment with your hair stylist
Target:	go to salon for appointment
Predicted:	drive to the barber shop

Table 2: Prediction examples.

Model	**Accuracy (%)**
BASELINE$_{NON-ATT}$	23.9
1-LAYER-BISEQ2SEQ$_{NON-ATT}$	**27.3**
2-LAYER-BISEQ2SEQ$_{NON-ATT}$	24.0
BASELINE$_{ATT}$	23.6
1-LAYER-BISEQ2SEQ$_{ATT}$	23.0
2-LAYER-BISEQ2SEQ$_{ATT}$	**25.5**

Table 3: The accuracy results of the LSTM-based models on the subset of 682 pairs.

i.e., belong to the same gold paraphrase set.

The accuracy results are given in Table 3 for the same LSTM-based models taken from Section 3.3. Accuracy is measured as the percentage of predicted sentences that occur *token-identical* in the paraphrase set of the corresponding target sentences. Our best model outperforms Pichotta and Mooney (2016)'s BASELINE by 3.4%.

Since the DeScript gold sets do not contain all possible paraphrases, an expert (computational linguist) checked cases of near misses between *Target* and *Predicted* (i.e. similar to the cases shown in Table 2) in a restrictive manner, not counting borderline cases. So we achieve a final average accuracy of about 31%, which is the sum of an average accuracy over 6 models in Table 3 (24%) and an average accuracy (7%) of checking cases of near misses (i.e, *Target* and *Predicted* are clearly event paraphrases).

The result does not look really high, but the task is difficult: on average, one out of 26 paraphrase sets (i.e., event types) per scenario has to be predicted, the random baseline is about 4% only. Also we should be aware that the task is *prediction of an unseen event*, not classification of a given event description. Continuations of a story are underdetermined to some degree, which implies that the upper bound for human guessing cannot be 100 %, but must be substantially lower.

4 Conclusions

In this paper, we explore the task of event prediction, where we aim to predict a next event addressed in a text based on the description of the preceding event. We created the new open-domain and closed-domain datasets based on WIKIHOW and DESCRIPT which are available to the public at: https://github.com/daiquocnguyen/EventPrediction. We demonstrated that more advanced SEQ2SEQ models with a bidirectional and multi-layer RNN architecture substantially outperform the previous work. We also introduced an alternative evaluation method for event prediction based on gold paraphrase sets, which focuses on semantic agreement between the target and predicted sentences.

Acknowledgments

This research was funded by the German Research Foundation (DFG) as part of SFB 1102 "Information Density and Linguistic Encoding." We would like to thank Hannah Seitz for her kind help and support. We thank anonymous reviewers for their helpful comments.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. Software available from http://tensorflow.org/.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 33–43.

Jeffrey L. Elman. 1990. Finding structure in time. *Cognitive Science*, pages 179–211.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, pages 1735–1780.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 3294–3302.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *Proceedings of the 4th International Conference on Learning Representations*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013*, pages 3111–3119.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318.

Karl Pichotta and Raymond J. Mooney. 2016. Using sentence-level lstm language models for script inference. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 279–289.

Ying Qin and Lucia Specia. 2015. Truly exploring multiple references for machine translation evaluation. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, pages 113–120.

M. Schuster and K.K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, pages 2673–2681.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, pages 1929–1958.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pages 3104–3112.

Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *Advances in Neural Information Processing Systems 28*, pages 2773–2781.

Lilian D. A. Wanzare, Alessandra Zarcone, Stefan Thater, and Manfred Pinkal. 2016. Descript: A crowdsourced corpus for the acquisition of high-quality script knowledge. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 3494–3501.