Fully-Convolutional Character-Level Seq2Seq for Dialogue Response Generation

Peter Henderson
peter.henderson@mail.mcgill.ca

Abstract

We present a character-level dialogue response generation system based on the ByteNet architecture proposed by Google DeepMind. We also modify the system to promote diversity and compare the system against a commonly used LSTM-based architecture with attention. We train both on the Cornell Movie Dialogue corpus and find that the two are comparable based on BLEU score and human judges. However, due to the small size and noisy nature of the Cornell Movie Dialogue corpus, neither the baseline nor the fully convolutional system yield particularly satisfying results. We suggest further work to train the system on a larger corpus and modifying the architecture further to generate more coherent and robust responses.

1 Introduction

Dialogue systems are a convenient paradigm for human-computer interfaces (HCIs). The mimicry of human interaction through conversation makes products more accessible to the general population. However, developing systems that can compete with human conversational intelligence is still an unsolved problem. Recent advances in statistical deep learning models have shown promising results toward human-level conversation (Vinyals and Le, 2015). These statistical models build a basis for more believable conversational systems by providing statistically relevant and appropriate responses learned from real human dialogue corpora.

Sequence-to-sequence (seq2seq) models take an input query, encode the query into a latent space and then use a decoder to generate an output sequence. It has been shown that seq2seq architectures using Long-Term Short-Term (LSTM) units can yield surprisingly effective results when trained on a large corpora (Vinyals and Le, 2015) and are common in state of the art statistical dialogue systems.

LSTMs (Hochreiter and Schmidhuber, 1997) are a subset of recurrent neural networks (RNNs) which take into account more context by using several "gates". These gates enable delayed updates at longer time-steps in addition to the short-term updates of a normal recurrent neural network. They take some input x and some state h such that the new state information is only updated with the long-term candidate information if a forget gate f activates a non-linearity. For the sake of brevity, see (Hochreiter and Schmidhuber, 1997; Bahdanau et al., 2014; Sutskever et al., 2014) for more details on LSTMs and seq2seq with RNNs.

Much work has been done in improving seq2seq models, typically used in machine translation tasks. An example of such improvements is an attention model, which has been added to the LSTM units to learn to "focus" on the proper keywords in the encoding sequence (Bahdanau et al., 2014).

Typically, these systems map words of a fixed vocabulary to an embedding space. While word-level embeddings currently produce state of the art results for machine translation and dialogue systems, recent work has also shown that character-level models can improve on machine-translation tasks (Kim et al., 2015; Kalchbrenner et al., 2016). These systems also have the benefit of not requiring a large vocabulary or dealing with unknown words.

A convolutional neural network (CNN) tries to replicate the human visual system by stacking layers of convolutional filters with learned weight parameters. These networks bring state of the art results in many visual tasks (Szegedy et al., 2015; Toshev and Szegedy, 2014; Krizhevsky et al., 2012). Furthermore, CNN architectures produce state of the art results in text-to-speech audio generation (Oord et al., 2016) and character-level machine translation (Kalchbrenner et al., 2016). The WaveNet and ByteNet architectures for text-to-speech and machine-translation tasks, respectively, additionally use dilated convolutions and causal convolutions. Dilated convolutions skip inputs at regular intervals such that the convolution takes into account longer contexts. Causal convolutions only take into account inputs at previous timesteps. This allows for the production of sequences which are coherent across longer decoding contexts. For brevity, a review of CNNs can be found in (Gu et al., 2015). More detailed descriptions of causal convolutions and dilated convolutions, as well as the WaveNet and ByteNet architectures can be found in (Oord et al., 2016; Kalchbrenner et al., 2016).

Due to the successes of the aforementioned paradigms, we expand on the work of character-level language modeling and apply a fully convolutional seq2seq system to the task of dialogue response generation. This is the first time a fully convolutional, character-level model has been applied to such a task to the authors' knowledge.

2 Method

2.1 Data

For training and evaluation of our models we use the Cornell Movie Dialogs Corpus (Danescu-Niculescu-Mizil and Lee, 2011). We choose this dataset because it is openly available and contains conversations in a wide array of domains tagged with character data. It is also a much smaller and more tractable dataset than alternatives such as OpenSubtitles (Lison and Tiedemann, 2016) (we found that training times on OpenSubtitles were unreasonable). The original dataset contains "220,579 conversational exchanges between 10,292 pairs of movie characters" according to the authors. We further prune this to query-response pairs of only 50 characters in length. Due the character-level nature of our system, extending this limit further increased training time by a significant amount. This resulted in around 89,000 total dialogue pairs. We further split this into a training set (80% of the data), validation set (10% of the data), and hold-out test set (10% of the data). We heavily modify a Python wrapper for the Cornell Movie Dialogue Corpus to download and process the data[1]. Special end-of-phrase characters (EOP) were appended to each query/response.

2.2 Baseline LSTM Seq2Seq with Attention

For our baseline, we use a similar seq2seq model as in (Vinyals and Le, 2015). The model uses 2 layers of LSTM units with the encoder and decoder each having hidden states of size 500. It uses an attention mechanism on the encoder as in (Bahdanau et al., 2014). For the sequence inputs and outputs, a vocabulary is made from the dataset and words are mapped to an embedding space of size 500, with backpropagation updating the embeddings during training. For the baseline Seq2Seq LSTM model, we use the Harvard NLP group implementation[2] without modification.

2.3 Baseline ByteNet

For our baseline fully convolutional model, we use the ByteNet architecture described in (Kalchbrenner et al., 2016). Characters are encoded into an embedding space of size 500 and input pairs are put in batches of size 16. The encoder of the model consists of 3 stacks of dilated convolution layers with kernel size 5 and dilation rates of 1, 2, 4, 8, and 16 (increasing at each layer and then repeating in each stack). The decoder consists of 3 stacks of dilated causal convolution layers with kernel size 3. The dilations in the decoder also increase at each layer from 1, 2, 4, 8, to 16 and then repeat at each stack. The layers are zero-padded to the maximum length sequence (50 characters). The output of the encoder is concatenated to the character embeddings of the target sequence and fed as inputs to the decoder. During evaluation, the next step predictions are populated and fed back into the system until an EOP character is reached. Batch normalization (Ioffe and Szegedy, 2015) is used on the encoding

[1] github.com/pralexa/SimpleBot

[2] github.com/harvardnlp/seq2seq-attn

layers and layer normalization (Ba et al., 2016) is used on the decoder. To avoid gradient vanishing, a residual connection is added at each layer similarly to Figure 1. For the sake of brevity, a more detailed explanation of the ByteNet model can be found in (Kalchbrenner et al., 2016). We heavily modify an open ByteNet implementation[3] implemented in TensorFlow (Abadi et al., 2015) with the SugarTensor wrapper[4] for our experiments.

2.4 Modified ByteNet

In statistical dialogue systems, there is a known problem where the default response is "I don't know". Li et al. suggest using a diversity-promoting objective function to tackle this problem (Li et al., 2015). Another alternative modifies the decoder architecture with a "glimpse" mechanism which shares attention between the encoder and decoder (Shao et al., 2017). For our approach, we modify the ByteNet architecture by adding global conditioning at every decoding layer based on the encoded latent state. This global conditioning is taken from the WaveNet architecture (Oord et al., 2016) and can be seen in Figure 1. The following equation mathematically describes the modified convolutional layer (W, V are weights, $W * x$ indicates a dilated convolution, $V * h$ is a 1x1 convolution to match the size of $W * x$, $\odot$ indicates point-wise multiplication).

$$z = \tanh(W * x + V * h) \odot \sigma(W * x + V * h)$$

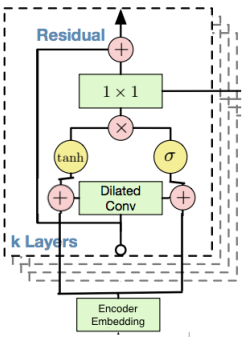

Figure 1: Residual block composing the modified ByteNet architecture. Image modified from (Oord et al., 2016).

[3] github.com/buriburisuri/ByteNet

[4] github.com/buriburisuri/sugartensor

For parity with the original ByteNet model, we keep all other parameters consistent.

2.5 Objective Function

We use Softmax Categorical Cross-Entropy as our objective function. This is consistent with several past works (Kalchbrenner et al., 2016; Vinyals and Le, 2015). For more information, please see the TensorFlow implementation (Abadi et al., 2015) or the previously cited works. We also experiment with L2 regularization (Ng, 2004). Initial findings seemed to produce more interesting and appropriate responses (by inspection) at ~30k gradient update steps, but after full convergence the model responded nonsensically. Consequently, we leave this out of our final analysis, but further tuning may improve responses.

2.6 Optimization Method

For the optimization method, we use MaxProp[5], as provided by the SugarTensor framework. According to its authors, MaxProp prevents gradient explosions without clipping heuristics and guarantees numerical stability. We also experimented using Adam with gradient-clipping (Kingma and Ba, 2014), but found that MaxProp yielded faster, more stable convergence. Convergence is characterized as no more loss improvements across 5 epochs (an epoch consisting of gradient updates on the full training set).

2.7 Beam Search

During evaluation we use basic beam search for all models as described in (Li et al., 2015; Li and Jurafsky, 2016). That is, we choose the highest probability sentence based on a greedy search consisting of k beams. We limit the search to 5 beams taking into account the top-5 probabilities at each timestep. More information on beam search can be found in (Reddy, 1977; Li et al., 2015; Li and Jurafsky, 2016).

2.8 Evaluation

For evaluation metrics, we use BLEU score as in (Li et al., 2015) and (Li et al., 2016). The implementation of BLEU score was taken from the Moses Translation Project [6]. Prior work (Liu et al., 2016) has argued that BLEU score is an ineffective metric as it does not account for a variety of possible

[5] See jamonglab.com/maxprop.

[6] github.com/moses-smt/mosesdecoder

	BLEU	Appropriate	Grammatical	Diverse
LSTM-W	1.30	2.16	**5**	1.45
BN-OC	1.46	**2.23**	4.59	1.72
BN-MC	**2.01**	1.89	4.17	**2.14**
Human	*X*	*4.68*	*5*	*4.26*

Table 1: BLEU score results of algorithms on hold out test-set of Cornell Movie Dialogs Corpus. Average Questionnaire results across 2 independent human evaluators. LSTM-W is the baseline LSTM word-level model. BN-OC is the character-level original ByteNet model. BN-MC is the modified character-level ByteNet.

responses. As such, we also use a "questionnaire" composed of 50 queries based on (Vinyals and Le, 2015) and (Shao et al., 2017). Two human judges, unfamiliar with the project, rated responses based on three criteria: (1) appropriateness (whether it addresses the query); (2) grammatical correctness; (3) diversity of the response/information gain (whether it provides more information than "I don't know"). Responses were rated on a scale from 1 (Bad) to 5 (Excellent). Human sample responses were also provided to the judges as part of the set of responses to rate.

3 Results, Discussion and Conclusion

The results of our experiments can be found in Table 1. We found that a character-level fully convolutional architecture can compete with the baseline LSTM seq2seq model, and even improves on BLEU score, appropriateness, and diversity. We posit that the detrimental effects on grammaticality are due to the character-level nature of the model and the small size of the dataset. Generally, neither model produces satisfying results, but we suggest that based on comparable results to the LSTM seq2seq model, similar results to (Vinyals and Le, 2015) may be obtained when training on larger corpora.

3.1 Response Coherence and Diversity

While our modified ByteNet model improves the diversity of responses from the baseline ByteNet and the baseline Seq2Seq model, this comes sometimes at the cost of grammaticality and appropriateness of the response as seen by the questionnaire results. We suggest that the impact on grammar may be attributed to the global conditioning gate outweighing the causal dependence on previous timesteps. Particularly, as the sentence grows longer, it is likely that less data has been seen at further timesteps (due to the average length of sentences being less than the maximum), therefore it relies purely on the global conditioning which results in non-grammaticality. Consequently, training on larger corpora or pre-training parts of the network for general language modeling may improve grammar scores.

3.2 Metrics, Data Quality, Training Time and Future Work

While we use BLEU score and a questionnaire to evaluate our system, the efficacy of these methods is still in debate. An overview of existing metrics can be found in (Liu et al., 2016). We suggest that the questionnaire is more appropriate than BLEU scores, as the results are based on human perception of the system (which is the goal for an HCI). ADEM (Lowe et al., 2017) is a promising alternative, but has not been reproduced or published.

Additionally, we find that the quality of the dataset to be suspect, which could be impacting results. In movie dialogues, there may be additional context necessary to understand a sequence between two characters. On observation, we found many query-response pairs that seemed nonsensical without the surrounding context. We posit that more data is required to smooth out any noise.

Overall, we wish to run our proposed algorithm on a larger corpus such as the Ubuntu Dialogue Corpus (Lowe et al., 2015), which we were unable to do due to long training times (some authors cite that even a 4-layer LSTM model takes around a month to converge on larger corpora (Li et al., 2016)). We also wish to experiment with further modifications to the architecture. While our preliminary changes showed promising results and were able to compete with a baseline LSTM model, it is possible that more changes (such as tuning L2 regularization or modifying the convolutional architecture) can be made to yield more appropriate and grammatical results without losing the diversity of our current model.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Cristian Danescu-Niculescu-Mizil and Lillian Lee. 2011. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*.

Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, and Gang Wang. 2015. Recent advances in convolutional neural networks. *arXiv preprint arXiv:1512.07108*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2015. Character-aware neural language models. *arXiv preprint arXiv:1508.06615*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Jiwei Li and Dan Jurafsky. 2016. Mutual information and diverse decoding improve neural machine translation. *arXiv preprint arXiv:1601.00372*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles.

Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *CoRR*, abs/1603.08023.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *arXiv preprint arXiv:1506.08909*.

Ryan Lowe, Michael Noseworthy, Iulian Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. Towards An Automatic Turing Test: Learning to Evaluate Dialogue Responses. In *Proceedings of the 34th International Conference on Machine Learning (ICML-17) (Pending Review)*.

Andrew Y Ng. 2004. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78. ACM.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

D Raj Reddy. 1977. Speech understanding systems: A summary of results of the five-year research effort.

Louis Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil. 2017. Generating Long and Diverse Responses with Neural Conversation Models. In *Proceedings of the 34th International Conference on Machine Learning (ICML-17) (Pending Review)*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich.

2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9.

Alexander Toshev and Christian Szegedy. 2014. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.