

Generative Adversarial Networks with Memory for Text Generation

Emily Sheetz

Overview

- Project Description
- Generative Adversarial Networks (GANs)
- Recurrent Network Architectures
- Learning Algorithms
- Tools and Project Timeline
- Socio-Technical Context

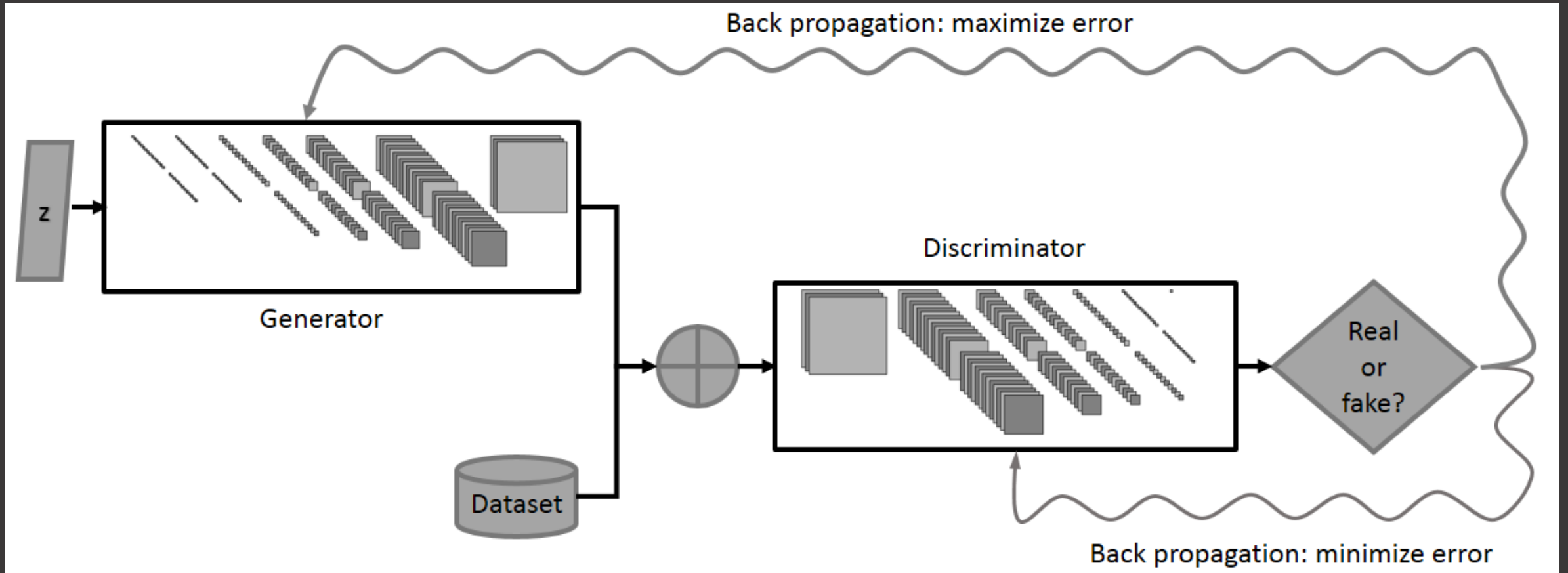
Project Description

- Train a neural network to generate sentences of horror text
- Generate samples at the character level
- Evaluate the quality of generated samples through survey
- Qualitative measurement

Generative Adversarial Networks (GANs)

- Framework containing two neural networks
- Generator (G) attempts to generate samples that could be from the dataset
- Discriminator (D) attempts to determine which samples are from the generator or from the data set

GAN Training Process



Heinrich, Greg. "Photo Editing with Generative Adversarial Networks (Part 1)." *NVIDIA, Parellel Forall* (2017). Accessed Aug. 2017.

Recurrent Neural Networks

- Generator and discriminator need memory for long-term structure
- Recurrent neural networks (RNNs) feed outputs from previous time steps back into network
- Output probability distribution of next element in sequence conditioned on previous elements

Recurrent Network Architectures

- RNNs do not train well within GAN framework
- Long Short-Term Memory Networks (LSTMs):
 - Selectively remember new information and forget information from previous time steps
 - Learn more complex long-term relationships than RNN
- Gated Recurrent Units (GRUs):
 - Comparable performance to LSTMs
 - Fewer parameters than LSTMs, meaning more computationally efficient

Work with Network Architectures

- GAN with GRUs for memory (Press et al.)
 - Python and TensorFlow
 - Problems running, but sample code for GAN
- TensorFlow tutorial with LSTMs
 - LSTMs for language modeling, predicting next word in sequence
 - Training time estimate, sample code for LSTM

Learning Algorithms

- Curriculum Learning: increase length of generated sequences
- Variable Length Outputs: allow for output sequences less than or equal to maximum length
- Teacher Helping: generator conditioned on samples in training data

Tools

- Project Gutenberg
- Python
- TensorFlow

Project Timeline

- Checkpoint 1: Script for text preprocessing
- Checkpoint 2: Run code from previous work, extend to horror text, architecture of network
- Checkpoint 3: Code GAN framework, LSTMs for memory
- Checkpoint 4: Code LSTMs, implement learning algorithms
- Checkpoint 5: Train GAN
- Checkpoint 6: Survey opinions on quality of text

Socio-Technical Context

- Humans tell stories to communicate
- Artificial intelligence given human-like intelligence through storytelling
- Horror elicits strong emotions
- Emotions of horror fiction depend on person
- AI participating in human tasks

Concluding Remarks

- GAN with LSTMs for memory
- Generate sentences of horror text
- Qualitative evaluation of quality of generated samples
- Human aspect of stories
- Artificial intelligence participating in human task of storytelling

Select Bibliography

- Goodfellow, Ian, et al. "Generative Adversarial Nets." *Advances in Neural Information Processing Systems*. 2014.
- Heinrich, Greg. "Photo Editing with Generative Adversarial Networks (Part 1)." *NVIDIA, Parallel Forall*, devblogs.nvidia.com/parallelforall/photo-editing-generative-adversarial-networks-1/. 2017. Accessed Aug. 2017.
- Horror Writers Association. "What is Horror Fiction?" *Horror Writers Association*, <http://horror.org/horror-is.htm>. 2009. Accessed Nov. 2017.
- Press, Ofir, et al. "Language Generation with Recurrent Generative Adversarial Networks without Pre-training." *arXiv preprint arXiv:1706.01399*. 2017.
- Rajeswar, Sai, et al. "Adversarial Generation of Natural Language." *arXiv preprint arXiv:1705.10929*. 2017.
- Riedl, Mark. "Why Artificial Intelligence Should Read and Write Stories." *Huffington Post*, www.huffingtonpost.com/mark-riedl/why-artificial-intelligen_b_8287478.html. 2015. Accessed Oct. 2017.
- TensorFlow. "Recurrent Neural Networks." *TensorFlow*, www.tensorflow.org/tutorials/recurrent. 2017. Accessed Oct. 2017.
- Xiao, Xuerong. "Text Generation using Generative Adversarial Training." *Stanford University*. 2017.