Cutting Down “Fluff”: A Twist on Text Summarization

Kate Salmon (ksalmon1@stanford.edu), Gustavo Torres da Silva (gdasilva@stanford.edu)

Motivation

Context
- People write every day.
- Getting rid of "fluff" (clutter in language) takes time.
- Text summarization is heavily explored, but not as an a way to assist people in writing more concisely.

Goal
Given a textual input, output a concise version of that text (i.e. keep the same information in fewer words).

Data

Approach
- No pre-built dataset mapping cluttered text to concise text
- We built our own
- Reverse-engineered from concise to fluffed:
  - Downloaded BBC dataset of news articles
  - Used a Text Inflator to add "fluff"
  - Original version => output
  - Fluffed version => input

BBC article: Even successful artists don’t think the lawsuits will benefit musicians. → Output

Fluffed version: Even successful artists don’t **particularly** think the lawsuits will benefit musicians. → Input

Stats
- Fluffed articles: avg. of 523 words (min. 122, max. 6,,208)
- Concise articles: avg of 384 words (min. 89, max. 4,432)

Split
The 2,225 articles were divided as follows:
- Train: 1,779 examples (80%)
- Development: 223 examples (10%)
- Test: 223 examples (10%)

Models

Basic LSTM (Extractive Approach)

- Inputs embedded fluffed text document with its associated binary labels based on the concise version of the text
- Outputs predictions for whether or not a given word in the fluffed document should appear in the concise version
- Probability that a word is labeled to be in the output text based on model predictions: **P >= 0.5**
- Based on the binary labels generated, the predicted concise text is generated and compared to the ground truth concise text using ROUGE score.

Loss: Binary Cross Entropy

$$BCE = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i)$$

Pointer-Generator (Abstractive Approach)

- Pointing: copies words from source text
- Generator: creates new sentences (seq2seq + attn)
- Bellow, p_gen is the probability of generating a word from the vocabulary (as opposed to copying).

$$p_{gen} = \sigma(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr})$$

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i = w} a_i^t$$

- Coverage: tracks content covered, avoids repetition.

$$c^t = \sum_{t'=0}^{t-1} a^{t'}$$

- Loss: finally, the loss combines the coverage loss with the original loss.

$$\text{loss}_t = -\log P(w_t^* + \lambda \sum_i \min(a_i^t, c_i^t))$$

Results

	ROUGE-1			ROUGE-2			ROUGE-L		
Model	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
Baseline	81.34	22.01	33.83	61.60	16.66	25.60	78.48	21.34	32.76
PG	60.67	75.49	60.88	55.69	68.11	55.33	59.14	72.53	58.94
PG + Cov	**94.98**	74.92	81.85	**90.21**	71.20	77.76	**94.12**	74.18	81.0
Basic LSTM	93.36	**90.63**	**91.97**	84.35	**81.87**	**83.09**	92.37	**89.67**	**90.99**

Table 1.ROUGE scores on different models..

Input	PG + Cov. Output	Reference
the film, showing out of competition in berlin, **basically** is nominated for three oscars, including **absolute** best actor for cheadle. sophie okonedo, who **kind of** plays cheadles wife tatiana, **basically** is nominated for **hardly** the best supporting actress **in a basically major way**. "even successful artists don't **particularly** think the lawsuits will benefit musicians," **generally** said report author mary wadden **, sort of contrary to popular belief**.	the film, showing out of competition in berlin , is nominated for three oscars, including **absolute** best actor for cheadle. sophie okonedo , who plays cheadles wife tatiana , is nominated for **hardly** the best supporting actress. even successful artists don't think the lawsuits will benefit musicians.	the film, showing out of competition in berlin, is nominated for three oscars, including best actor for cheadle. sophie okonedo, who plays cheadle's wife tatiana, is nominated for best supporting actress. even successful artists don't think the lawsuits will benefit musicians , " **said report author mary madden**.

Table 2. Example inputs, outputs and references, highlighting the strengths and weaknesses of the Pointer-Generator model.

Input	LSTM	Reference
Ferguson hails Man Utds **really** resolve Manchester Uniteds Alex Ferguson literally has **for all intents and purposes** praised his players gutsy performance in the 10 generally win at Aston Villa, **which is fairly significant**. That **generally** was our **almost** the hardest away game of the season and it was a fantastic game of football, endtoend with lots of **pretty** good passing, said the Old Trafford boss,	ferguson hails man utds resolve manchester uniteds alex ferguson has **for all** praised his players gutsy performance in the 10 win at aston villa **which is** that was our **almost** the away game of the season it was fantastic game of football endtoend with lots of passing said the old trafford boss	ferguson hails man utds resolve manchester uniteds alex ferguson has praised his players gutsy performance in the 10 win at aston villa that was our hardest away game of the season and it was a fantastic game of football endtoend with lots of good passing said the old trafford boss

Table 3. Example inputs, outputs and references, highlighting the strengths and weaknesses of the LSTM model.

Discussion

Extractive
- Identifies large portion of **“fluff”** expressions.
- Removes most expressions , but keeps chopped versions of some: **ex. which is quite significant** becomes **which is** and **for all intents and purposes** becomes **for all**.
- Output of the text is a little sloppy (no punctuation) but does a decent job for out specific problem overall.
- LSTM units and the maximum output length were key parameters to define (experiment more with these if had more time).
- Model training is very slow due to the length of input and outputs.

Pointer Generator
- Identifies a large portion of **"fluff"** expressions.
- Yet, still **keeps some**.
- Coverage system fixed repetition issues, but not entirely. Lots of repetition in short passages. Model may assume minimum length.
- Occasionally **removed information** from the input.
- Model training is very slow due to length of input/output (in fig 1, only the steps in the **red box** were over our dataset -- the other ones were from the pretrained model).

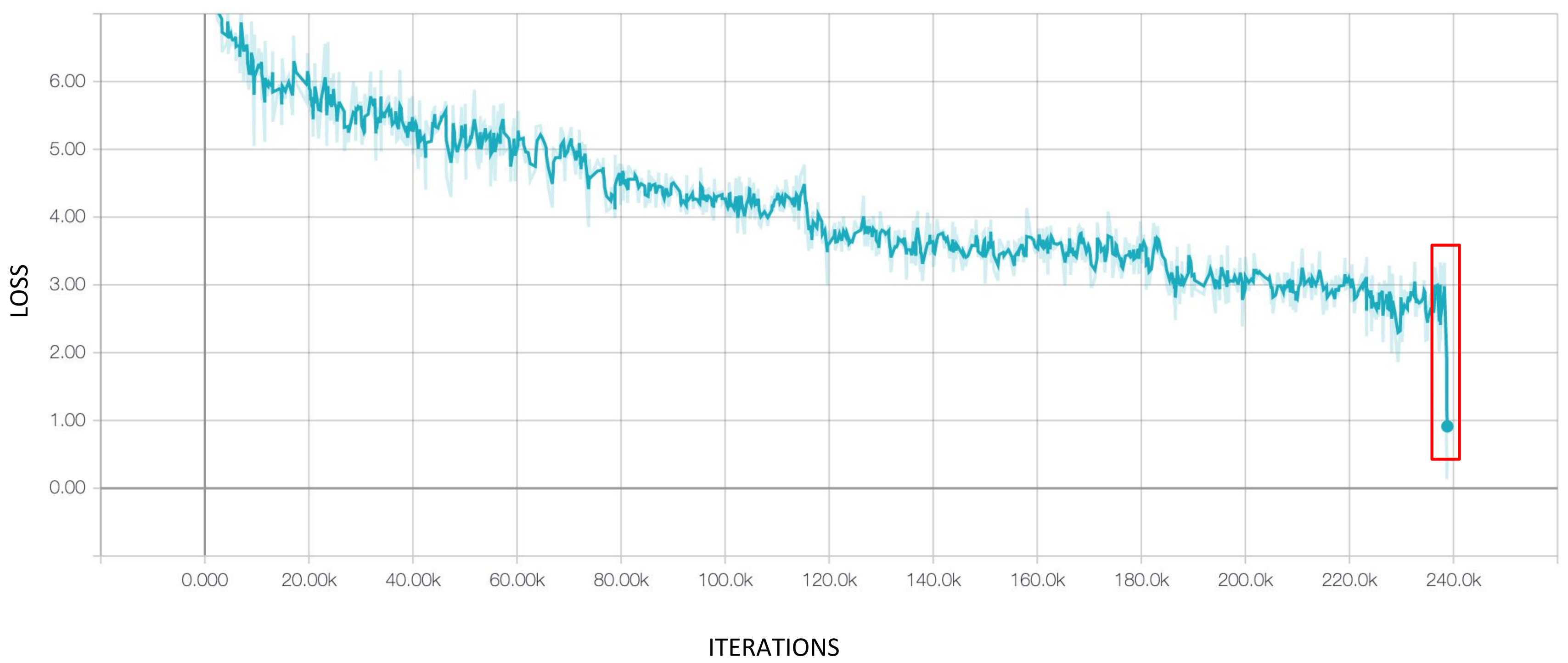


Fig. 1. Training loss X Iterations for the Pointer-Generator + Coverage model.

Future Work

Data: build dataset with other types of fluff.

Model experimentation: train and test a wider variety of summarization models.

Performance: explore ways to deal with longer outputs without incurring into huge time costs.

References

[1] See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

[2] Narayan, S., Cohen, S. B., & Lapata, M. (2018). Ranking sentences for extractive summarization with reinforcement learning. *arXiv preprint arXiv:1802.08636*.

[3] Cheng, J., & Lapata, M. (2016). Neural summarization by extracting sentences and words. *arXivpreprint arXiv:1603.07252*.

[4] Zhong, M., Liu, P., Wang, D., Qiu, X., & Huang, X. (2019). Searching for Effective Neural Extractive Summarization: What Works and What’s Next. *arXiv preprint arXiv:1907.03491*.

[5] Y. Zhang et. al.Extractive Document Summarization Based on Convolutional Neural Networks,IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society p. 918-922,2016.

Cutting Down "Fluff": A Twist on Text Summarization

Kate Salmon (ksalmon1@stanford.edu), Gustavo Torres da Silva (gdasilva@stanford.edu)

Motivation

Context

- People write every day.
- Getting rid of "fluff" (clutter in language) takes time.
- Text summarization is heavily explored, but not as an a way to assist human writing.

Goal

Given a textual input, output a concise version of that text (i.e. keep the same information in fewer words).

Data

Approach

- No pre-built dataset mapping cluttered text to concise text
- We built our own
- Reverse-engineered from concise to fluffed:
  - Downloaded BBC dataset of news articles
  - Used a Text Inflator to add "fluff"
  - Original version => output
  - Fluffed version => input

BBC article: Even successful artists don't think the lawsuits will benefit musicians. → Output

Fluffed version: Even successful artists don't **particularly** think the lawsuits will benefit musicians. → Input

Stats

- Fluffed articles: avg. of 523 words (min. 122, max. 6,,208)
- Concise articles: avg of 384 words (min. 89, max. 4,432)

Split

The 2,225 were divided as follows:

- Train: 1,778 examples (80%)
- Development: 223 examples (10%)
- Test: 223 examples (10%)

Models

Basic LSTM (Extractive Approach)

- Inputs embedded fluffed text document with its associated binary labels based on the concise version of the text
- Outputs predictions for whether or not a given word in the fluffed document should appear in the concise version
- Probability that a word is labeled to be in the output text based on model predictions: **P >= 0.5**
- Based on the binary labels generated, the predicted concise text is generated and compared to the ground truth concise text using ROUGE score.

Loss: Binary Cross Entropy

$$BCE = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot log(\hat{y}_i) + (1-y_i)\cdot log(1-\hat{y}_i)$$

Pointer-Generator (Abstractive Approach)

- Pointing: copies words from original
- Generator: creates new sentences (seq2seq + attn)
- Bellow, p_gen is the probability of generating a word from the vocabulary (as opposed to copying).

$$p_{\text{gen}} = \sigma(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr})$$

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_i^t$$

- Coverage: tracks content covered, avoids repetition.

$$c^t = \sum_{t'=0}^{t-1} a^{t'}$$

- Loss: finally, the loss combines the coverage loss with the original loss.

$$\text{loss}_t = -\log P(w_t^*) + \lambda \sum_i \min(a_i^t, c_i^t)$$

Results

	ROUGE-1			ROUGE-2			ROUGE-L		
Model	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
Baseline	81.34	22.01	33.83	61.60	16.66	25.60	78.48	21.34	32.76
PG	60.67	75.49	60.88	55.69	68.11	55.33	59.14	72.53	58.94
PG + Cov	**94.98**	74.92	81.85	**90.21**	71.20	77.76	**94.12**	74.18	81.0
Basic LSTM	93.36	**90.63**	**91.97**	84.35	**81.87**	**83.09**	92.37	**89.67**	**90.99**

Table 1.ROUGE scores on different models..

Input	PG + Cov. Output	Reference
the film, showing out of competition in berlin, **basically** is nominated for three oscars, including **absolute** best actor for cheadle. sophie okonedo, who **kind of** plays cheadles wife tatiana, **basically** is nominated for **hardly** the best supporting actress **in a basically major way**	the film, showing out of competition in berlin , is nominated for three oscars, including **absolute** best actor for cheadle. sophie okonedo , who plays cheadles wife tatiana , is nominated for **hardly** the best supporting actress.	the film, showing out of competition in berlin, is nominated for three oscars, including best actor for cheadle. sophie okonedo, who plays cheadle's wife tatiana, is nominated for best supporting actress.
"even successful artists don't **particularly** think the lawsuits will benefit musicians," **generally** said report author mary wadden **, sort of contrary to popular belief**.	even successful artists don't think the lawsuits will benefit musicians.	even successful artists don't think the lawsuits will benefit musicians , " **said report author mary madden**.
"even successful artists don't **particularly** think the lawsuits will benefit musicians," **generally** said report author mary wadden **, sort of contrary to popular belief**.	among those expected to receive nominations are the aviator , million dollar baby and sideways . academy members will now vote for their favourites before the final nominees are announced on 25 january .	among those expected to receive nominations are the aviator , million dollar baby and sideways . academy members will now vote for their favourites before the final nominees are announced on 25 january .

Table 2. Example inputs, outputs and references, highlighting the strengths and weaknesses of the model.

Discussion

Extractive

- Identifies large portion of **"fluff"** expressions
- Removes most expressions , but keeps chopped versions of some: **ex. which is quite significant** becomes **which is**
- Output of the text is a little sloppy, (no punctuation) but does a decent job for out specific problem overall

Input	Output Basic LSTM	Reference
Ferguson hails Man Utds **really** resolve Manchester Uniteds Alex Ferguson **literally** has **for all intents and purposes** praised his players gutsy performance in the 10 generally win at Aston Villa,**which is fairly significant**. That **generally** was our **almost** the hardest away game of the season and it was a fantastic game of football, endtoend with lots of **pretty good** passing, said the Old Trafford boss.	ferguson hails man utds resolve manchester uniteds alex ferguson has **for all** praised his players gutsy performance in the 10 win at aston villa **which is**that was our **almost** the away game of the season it was fantastic game of football endtoend with lots of passing said the old trafford boss.	ferguson hails man utds resolve manchester uniteds alex ferguson has praised his players gutsy performance in the 10 win at aston villa that was our hardest away game of the season and it was a fantastic game of football endtoend with lots of good passing said the old trafford boss.

Pointer Generator

- Identifies a large portion of **"fluff"** expressions.
- Yet, still **keeps some**.
- Coverage system fixed repetition issues, but not entirely. Lots of repetition in short passages. Model may assume minimum length.
- Occasionally **removed information** from the input.
- Model training is very slow due to length of input/output (in fig 1, only the steps in the **red box** were in our dataset -- the other ones were from the pretrained model).

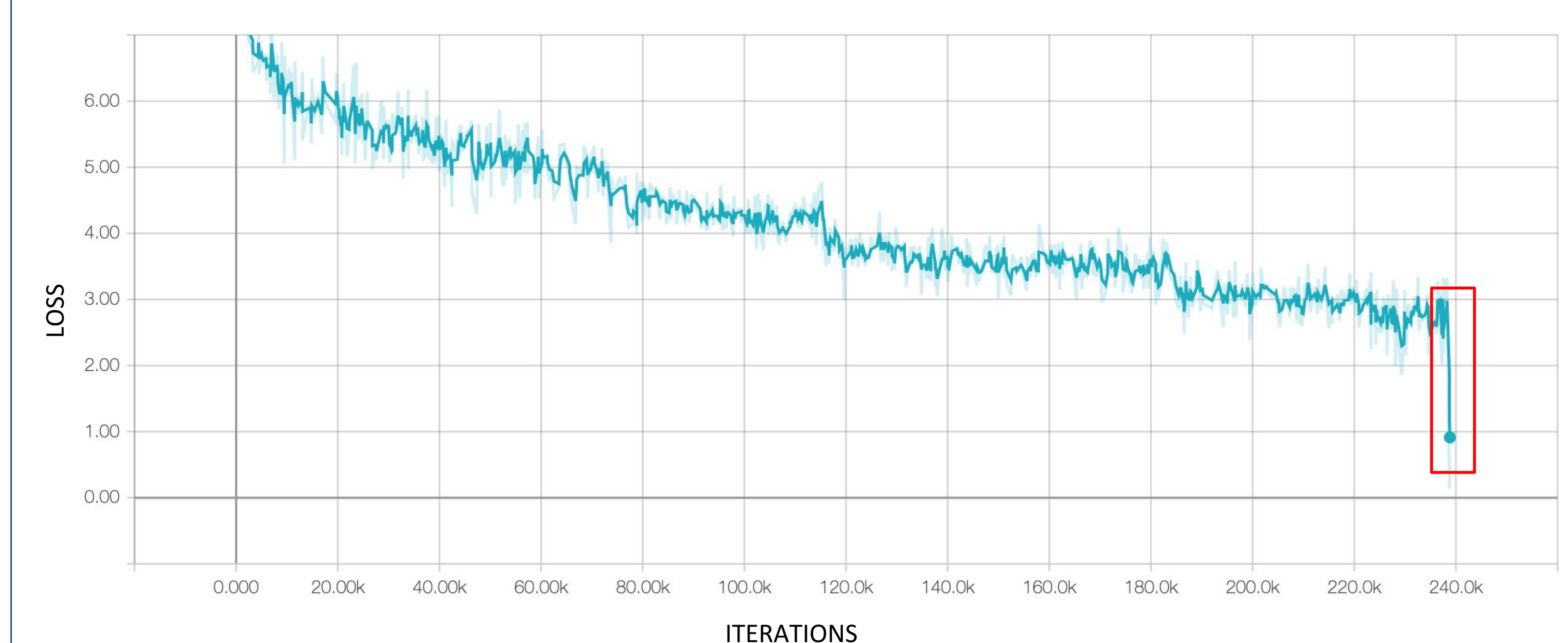


Fig. 1. Training loss X Iterations.

Future Work

Data: build dataset with other types of fluff.

Model experimentation: train and test a wider variety of summarization models.

Performance: explore ways to deal with longer outputs without incurring into huge time costs.

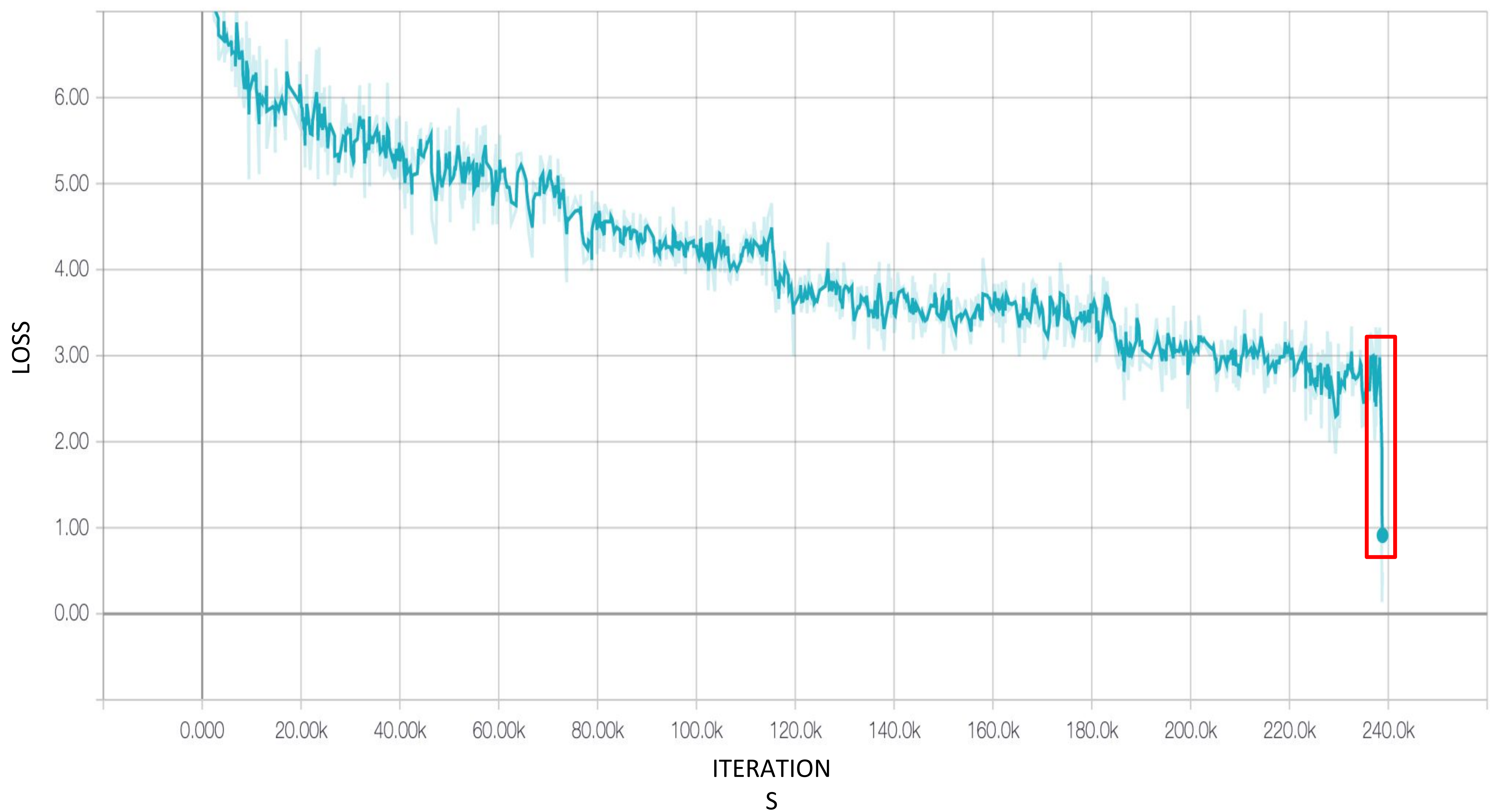
References

[1] See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

[2] Narayan, S., Cohen, S. B., & Lapata, M. (2018). Ranking sentences for extractive summarizationwith reinforcement learning. *arXiv preprint arXiv:1802.08636*.

[3] Cheng, J., & Lapata, M. (2016). Neural summarization by extracting sentences and words. *arXivpreprint arXiv:1603.07252*.

[4] Liu, Y., & Lapata, M. (2019). Text summarization with pretrained encoders. *arXiv preprintarXiv:1908.08345*.


LOSS
6.00
5.00
4.00
3.00
2.00
1.00
0.00
0.000
20.00k
40.00k
60.00k
80.00k
100.0k
120.0k
140.0k
160.0k
180.0k
200.0k
220.0k
240.0k
ITERATION
S

Cutting Down "Fluff": A Twist on Text Summarization

Kate Salmon (ksalmon1@stanford.edu), Gustavo Torres da Silva (gdasilva@stanford.edu)

Motivation

- People write every day.
- Getting rid of language clutter takes time.
- Text summarization is heavily explored, but not as an a way to assist human writing.
- **Goal: given a textual input, output a concise version of that text keeping the same information in fewer words.**

Data

Approach

- No pre-built dataset mapping cluttered text to concise text
- So we built our own
- Reverse-engineered from concise to fluffed:
  - Downloaded BBC dataset of news articles
  - Used a Text Inflator to add "fluff" to it
  - Original version => output
  - Fluffed version => input

BBC article: Even successful artists don't think the lawsuits will benefit musicians. → Output

Fluffed version: Even successful artists don't **particularly** think the lawsuits will benefit musicians. → Input

Stats

- Fluffed articles: avg. of 523 words (min. 122, max. 6,,208)
- Concise articles: avg of 384 words (min. 89, max. 4,432)

Split

- Train: 1,778 examples (80% of dataset)
- Development: 223 examples (10% of dataset)
- Test: 223 examples (10% of dataset)

Models

Abstractive

Results

	ROUGE-1			ROUGE-2			ROUGE-L		
Model	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
Baseline	0.8134	0.2201	0.3383	0.6160	0.1666	0.2560	0.7848	0.2134	0.3276
PG	0.6067	0.7549	0.6088	0.5569	0.6811	0.5533	0.5914	0.7253	0.5894
PG + Cov	0.9498	0.7492	0.8185	0.9021	0.7120	0.7776	0.9412	0.7418	0.8107

Input	Output	Reference
the film, showing out of competition in berlin, **basically** is nominated for three oscars, including **absolute** best actor for cheadle. sophie okonedo, who **kind of** plays cheadles wife tatiana, **basically** is nominated for **hardly** the best supporting actress **in a basically major way**	the film , showing out of competition in berlin , is nominated for three oscars , including **absolute** best actor for cheadle. sophie okonedo , who plays cheadles wife tatiana , is nominated for **hardly** the best supporting actress.	the film, showing out of competition in berlin , is nominated for three oscars, including best actor for cheadle. sophie okonedo , who plays cheadle's wife tatiana, is nominated for best supporting actress.
"even successful artists don't **particularly** think the lawsuits will benefit musicians," **generally** said report author mary wadden **, sort of contrary to popular belief**.	even successful artists don't think the lawsuits will benefit musicians .	even successful artists don't think the lawsuits will benefit musicians , " **said report author mary madden**.
among those expected to mostly receive nominations **basically** are the aviator, million dollar baby and sideways **in a basically big way** academy members will now vote for their favourites before the final nominees **for the most part** are announced on 25 january **n a actually major way**.	among those expected to receive nominations are the aviator , million dollar baby and sideways . academy members will now vote for their favourites before the final nominees are announced on 25 january .	among those expected to receive nominations are the aviator , million dollar baby and sideways . academy members will now vote for their favourites before the final nominees are announced on 25 january .

Discussion

Future Work

References

TITLE

Author1 (author1@stanford.edu), Author 2 (author@stanford.edu)

Motivation

Data

Models

Results

Discussion

Future Work

References

Predicting Satellite Images Using Spatiotemporal Generator Networks

Kristine Guo (kguo98@stanford.edu), Gustavo Torres (gdasilva@stanford.edu), Blanca Villanueva (villanue@stanford.edu)

Motivation

Motivation

- Wide variety of applications from gathering insights on climate change to analyzing economic development
- **Goal: predict environmental changes utilizing satellite images in a time sequence**

Problem Definition

For window size *t*, train a model *f* to produce an output image $\hat{x}^{(i)}_{t+1}$ that estimates the true *(t+1)*'th image $x^{(i)}_{t+1}$:

$$f : x^{(i)} = \{x_0^{(i)}, x_1^{(i)}, ..., x_t^{(i)}\} \to \hat{x}_{t+1}^{(i)}$$

Data

Sentinel-2 data are organized around regions referred to as **tiles**, 10980×10980 px images (10m/px) with a 5-day window between each still image. Satellite images in the Sentinel-2 dataset capture different perspectives of the same region depending relative orbit (Figure 1).

Pre-processing: Downsample to 5490×5490 px and split into crops of size **256×256 px**.

Creating time sequences: Use {1, ..., t} images with the same relative orbit to predict the (t + 1)'th image.

Our dataset: Paired dataset with: 2824 training (28240 images), 352 val (3520 images), and 352 test (3520 images) examples.

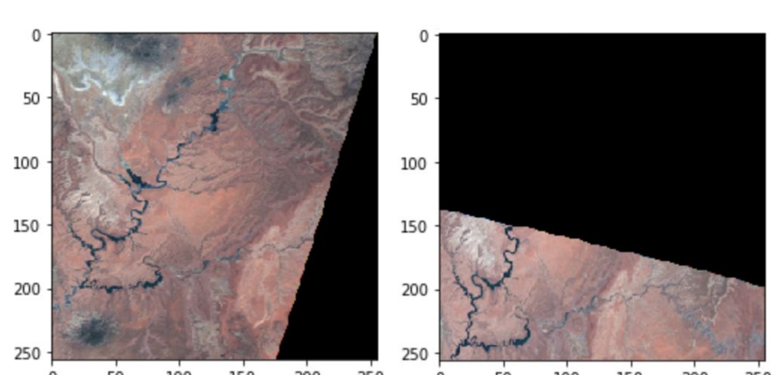


Figure 1. The same tile captured at two different relative orbits.

Evaluation

- We measure our results by comparing the ground-truth images at time (t + 1) with the model-generated images.
- **SSIM Index.** Measures the similarity between two images using perceived change in structural information.
- **PSNR Score.** Measure the quality of the reconstructed (or generated) image.

Models

Baseline

Given a temporal sequence of satellite images, we compute the images' mean and use this as the prediction for the next timestamp's image.

Pix2pix

Pix2pix uses conditional adversarial networks to implement image-to-image translation. We have:

- the generator G, that tries to generate an image that looks similar to images from the target domain y;
- the discriminator D, that learns to distinguish fake examples generated by G from real ones in y.

Loss function:

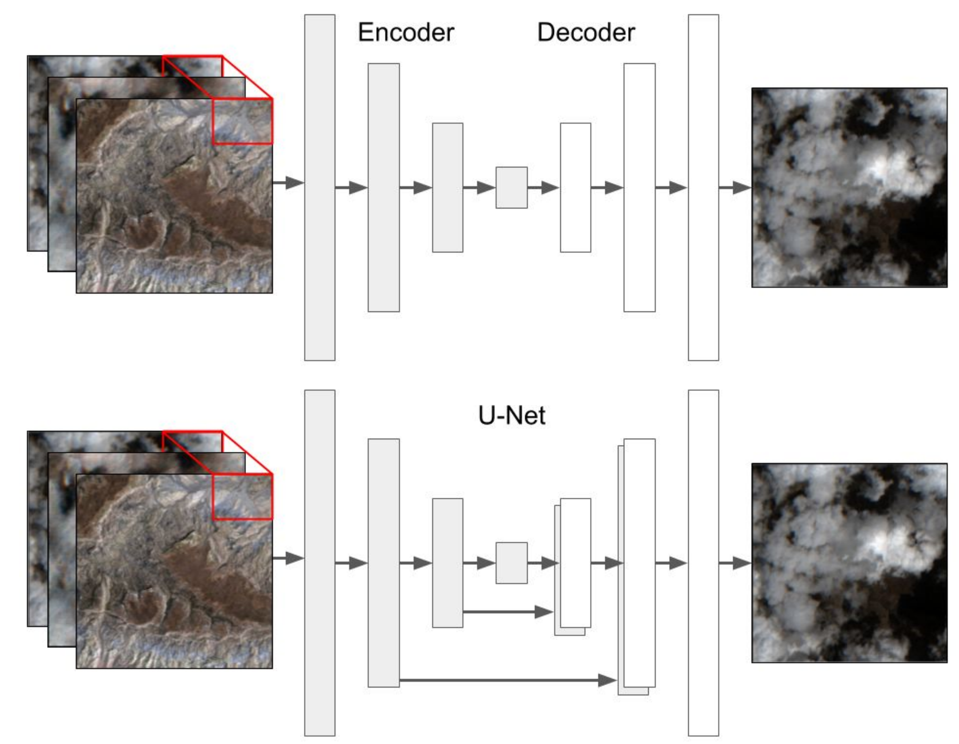
$$\begin{aligned}(G, D) =& \mathbb{E}_{x,y}[\log D(x, y)] + \\ & \mathbb{E}_{x,z}[\log(1 - D(x, G(x, z)))] \cdot \\ & \lambda \cdot \mathbb{E}_{x,y,z}[||y - G(x, z)||_1]\end{aligned}$$

We aim to solve:

$$G^* = \arg\min_G \max_D \mathcal{L}(G, D)$$

Architecture

We use the original Pix2pix architecture,whose modules are all of the form Convolution-Batchnorm-ReLU.



U-Net looks just like an Encoder-Decoder, except that it includes skip connections to transmit low-level information shared between input and output.

Results

	Mean SSIM (val)	Mean SSIM (test)	Mean PSNR (val)	Mean PSNR (test)
Baseline (t=1, Clear)	0.489	0.494	18.5	18.9
Baseline (t=1, Cloudy)	0.401	0.410	15.6	15.5
Baseline (t=5, Clear)	0.518	0.525	18.1	18.4
Baseline (t=5, Cloudy)	0.427	0.429	15.4	15.3
Pix2Pix (t=5, Clear)	0.521	**0.534**	19.7	**20.2**
Pix2Pix (t=5, Cloudy)	0.477	0.480	17.9	18.0
Baseline (t=8, Clear)	0.428	0.450	15.2	15.2
Baseline (t=8, Cloudy)	0.419	0.429	14.8	14.6
Pix2Pix (t=8, Clear)	0.429	0.447	16.7	17.2
Pix2Pix (t=8, Cloudy)	0.462	0.469	17.6	17.3

Table 1. Average SSIM and PSNR scores for all tested models. Pix2pix models were trained for 60 epochs.

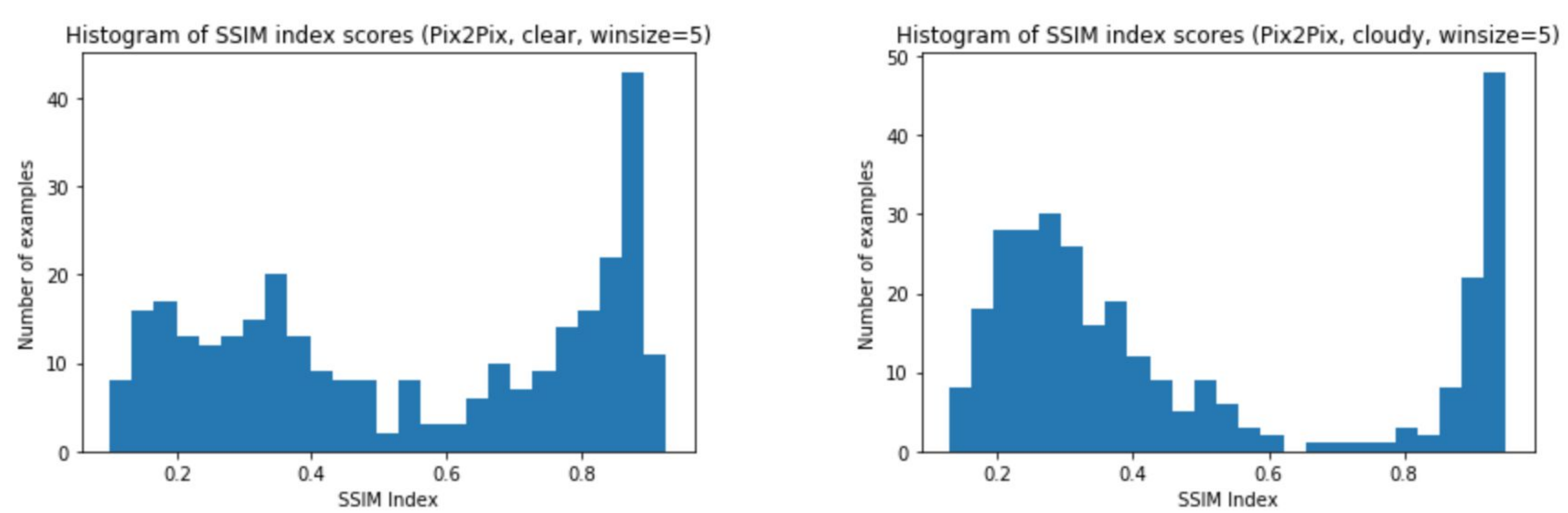
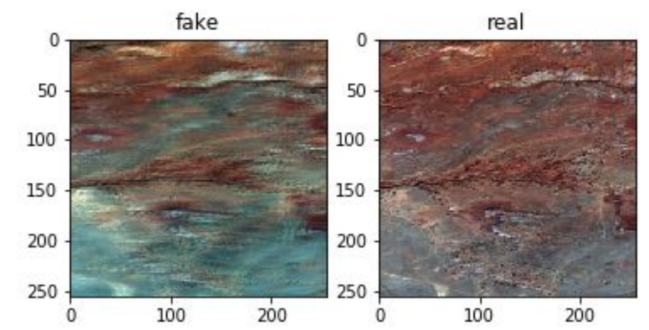


Figure 2. Histograms of the SSIM scores for all synthetic-real image pairs for the Pix2pix model trained on the clear (left) and cloudy (right) datasets with window size t=5.

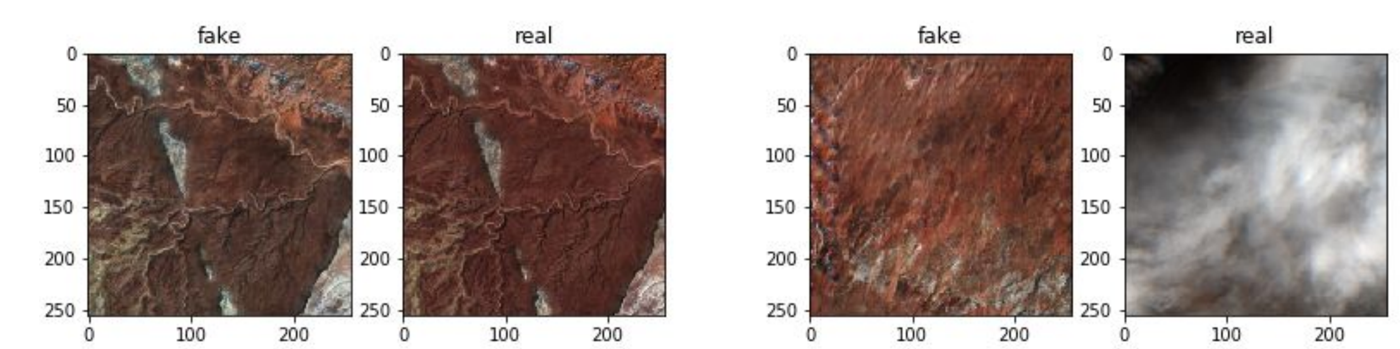
Discussion

Observations

- Even a simple baseline performs well, likely due to the staticity of landforms and the high temporal resolution
- Pix2pix outperforms the baseline for cloudy and non-cloudy images by mean SSIM and PSNR
- The wide performance gap between Pix2pix and baseline on cloudy data implies that GANs are able to capture more challenging mapping from input to output
- Tradeoff with window size
  - Baseline: large window sizes may result in excess noise (see right)



- Main weakness of the models: limited ability to predict the amount and structure of cloud coverage in examples with volatile weather
  - Distribution of SSIM scores for the Pix2pix model trained on the non-cloudy dataset is more right-skewed than the distribution of the Pix2pix model trained on the cloudy dataset (see below)
  - Short-term weather (such as extra cloud coverage) cannot always be safely extrapolated to future predictions (see below)
- Many Pix2pix-generated images look qualitatively similar to the target; however, the SSIM for these images can still be low
  - Visual structure of the generated clouds heavily impacts the SSIM



Contributions

- Applied GANs to challenging examples (e.g., volatile weather) and achieved results comparable to recent literature [1]
- Created a paired dataset of low-resolution images, since hi-resolution images are expensive and not publicly available

Future Work

Data: Collect more data across a wider variety of lat/lon/time.

Temporal resolution: Prediction across larger time intervals should be less susceptible to volatile and short-term weather patterns and predict larger geological trends.

Pix2pix with 3D convolutions: Explicitly incorporate temporal information.

References

[1] B. U. S. E. Vishnu Sarukkai, Anirudh Jain. Cloud removal in satellite images using spatiotemporal generative networks. 2019

[2] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. CoRR, abs/1611.07004, 2016.

[3] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4):600–612, April 2004.