# Transformers and Pointer-Generator Networks for Abstractive Summarization

Jon Deaton, Austin Jacobs, and Kathleen Kenealy
{jdeaton, ajacobs7, kkenealy}@stanford.edu

## Overview and Motivation

- Transformers have outperformed RNNs on many seq2seq tasks - including abstractive summarization
- However, transformers produce many of the same issues as RNN-based models do in summarization:
  - Repetition
  - Factual inaccuracies
- We present variations of transformers that use techniques that successfully addressed these issues in RNNs

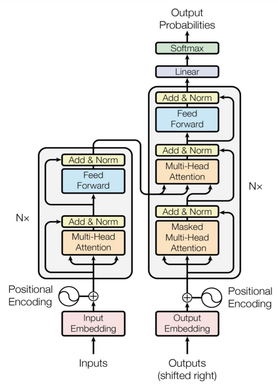
## Data

- We use the CNN/DM dataset:
  - 300,000 pairs of news articles and multi-sentence summaries
  - We used 40,000 pairs for training, 11,000 for validation
- Weaknesses: not ideal for summarization - originally meant for Question Answering
- Widely for summarization; good basis for comparison with other models.

## Approaches

**Transformer**

For our baseline, we use an open-source implementation of the basic transformer architecture as laid out in the seminal paper *Attention is all you need* [2].




## Approaches

**N-Gram Blocking**

- Used to reduce repetition
- During beam search, eliminates words that create an n-gram that already exists in the summary

**Coverage Loss**

- Also used to reduce repetition
- Create a coverage vector $c^t$, given all attention distributions over previous time steps:

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

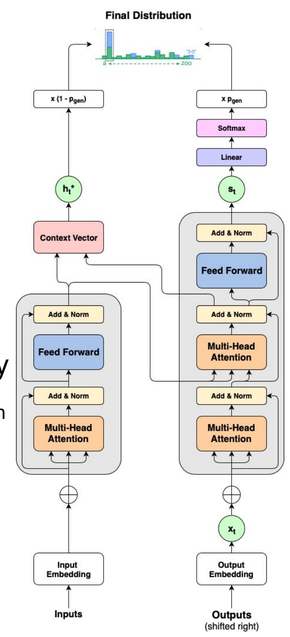
- Use to define the coverage loss, which gets added to the final loss of the transformer with a weight of λ

$$loss_t = -\log P(w_t^*) + \lambda \sum_i \min(a_i^t, c_i^t)$$

**Pointer Generator Network**

- Used to reduce factual inaccuracies and OOV mishandling
- Uses final joint attention to copy words from the source text with learned probability 1 - $p_{gen}$ where $p_{gen}$ is calculated as follows

$$p_{gen} = \sigma\left(w_h^\top h_t^* + w_s^\top s_t + w_x^\top x_t + b_{ptr}\right)$$




## Results

| Model | Rouge-1 |
|---|---|
| Transformer | 20.23 |
| Transformer + Pointer-Generator | 22.10 |
| Transformer + Pointer-Generator + Coverage | 22.93 |
| Transformer + Pointer-Generator + N-Gram Blocking (2-grams) | 25.31 |

**Example Summaries**

**Human Summary:** "oleg kalashnikov died of gunshot wounds, ukraine's interior ministry said. he was a party of regions deputy in ukraine's previous parliament. kalashnikov was ally of deposed ukrainian president viktor yanukovych."

**Baseline Transformer Summary:** "viktor [UNK] was found dead at the vladimir putin's death. he was found dead in his remained remained remained remained remained ..."

**Pointer-Generator Summary:** "russian oleg oleg oleg oleg sergei yanukovych was found dead. viktor was found dead at home in kiev in kiev. viktor yanukovych was found dead in the the kremlin has been been been been since."

**Pointer-Generator + Coverage Summary:** "ukrainian prime minister yanukovych was found dead in his home in kiev. ukrainian leader was found dead in his home in kiev, in kiev."

**Pointer-Generator + 2-Gram Summary:** "oleg viktor yanukovych was found dead in his home in kiev. he is of the ukraine of ukraine's parliament. the former ukrainian president vladimir putin says he was `` ``the communist party". "

## Discussion

- Baseline: transformer scores, while lower than expected, were high considering our limited training time: we achieved ⅔ the score of similar transformers in ½ the time [3]
- The Pointer-Generator network significantly improves performance; again, additional training time could improve these scores
- As expected, coverage slightly improved our ROUGE scores; however, we now face repetition of phrases and ideas rather than repetition of individual words
- N-gram blocking produces the largest increase in ROUGE scores and also produces that most fluid, realistic summaries

## Future Work

- Run all models and variants for extended periods of time
- Try several transformer model variants:
  - More layers
  - Different ways of calculating the final attention distribution
- Further experimentation with model restoration/hyperparameter tuning for coverage loss
- Testing with additional datasets (New York Times, Gigaword, etc.)

## References

[1] Abigail See and Peter J. Liu and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073-1083, 2017

[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *In the Annual Conference on Neural Information Processing Systems (NIPS)*, 2017

[3] Nima Sanjabi. Abstractive text summarization with attention-based mechanism. Master's thesis, Universitat Politècnica de Catalunya, July 2018.