

## TODO due FRIDAY:

### Zakhar

- train generator of size 256
- draw diagram of our generator and discriminator
- help Dorka with a description of models in the report

### Michael

- train generator of size 128

### Dorka

- train generator of size 512
- describe formally models in the report

### Danko

- collect all the results from the models above
- merge Michael's and Danko's files and add discriminator training to the main Colab notebook
- start training discriminator
- informal description of generator and discriminator (done below)

## GENERATOR

- We use Seq2Seq model, or in other words, Encoder-Decoder model; kinda well described here <https://arxiv.org/pdf/1602.06023.pdf> with a good illustration
- **Encoder:**
  - o Embedding layer (we use pretrained 200d GloVe) -> Dropout -> Bidirectional GRU with one layer and a hidden size of 128/256/512
  - o RNN return output and hidden state. The last hidden state (both from forward and backward RNN) is further fed through fully-connected layer and hyperbolic-tangent non-linearity
- **Decoder with attention:**
  - o attention is described in NLP Lecture 11
  - o Unidirectional GRU RNN with the same size as of encoder one
- for more information follow our code here [https://github.com/stanclid/GeneratingHeadline\\_GANs/blob/master/Code/Models/Attention\\_seq2seq.py](https://github.com/stanclid/GeneratingHeadline_GANs/blob/master/Code/Models/Attention_seq2seq.py)

## DISCRIMINATOR

- We use exactly the model described in <https://arxiv.org/pdf/1408.5882.pdf> (there are good illustrations and descriptions)
- We have 3 convolutional layers with a filter size of 3, 4 and 5; there are always XY (TBD) kernel filters. This layer is followed by ReLU non-linearity and 1d MaxPool. This maxpool is through the whole sequence. (please see our code)

[https://github.com/stanclD/GeneratingHeadline\\_GANs/blob/master/Code/Models/CNN\\_text\\_clf.py](https://github.com/stanclD/GeneratingHeadline_GANs/blob/master/Code/Models/CNN_text_clf.py) )

- All the convolutions are applied directly on input sentence (summary) and their outputs after an application of ReLu and MaxPool are then concatenated into a vector.
- Then, this is followed by dropout layer, fully-connected layers and softmax non-linearity giving output probabilities indicating whether a summary is real or generated