Global and Local Critical Policy Learning for Abstractive Summarization

Gautier Dagan
University of Amsterdam
gautier.dagan@student.uva.nl

Alexander Geenen
University of Amsterdam
geenen124@gmail.com

Diana Rodriguez
University of Amsterdam
diana.rodrigzluna@gmail.com

Gururaja Rao
University of Amsterdam
gururajraop@gmail.com

ABSTRACT

Traditional Abstractive Summarization models suffer from training directly using a maximum likelihood approach, which is known to decrease the models' abstractive power and therefore generate summaries which are less human-like in abstraction. Our approach attempts to solve this by incorporating a *local* (word level) and *global* (sentence level) loss weighting using the ROUGE metric directly as a reward in a Policy Gradient. By evaluating sub-sequences of the generated summary, we can obtain the gain they provide in ROUGE score for the entire summary, and weigh the loss locally and globally to reflect this.

Our new loss and training procedure pushes the networks closer to the golden reference summaries and allows us to optimize for ROUGE score and increase the abstractiveness of the generated summaries. Reweighing loss by the local and global rewards give the generator a better understanding of what parts of the generated summary are better than others. We are able to considerably surpass a baseline generator trained using MLE, by using a mix of our local and global approaches with more emphasis on local rewards. We expect that using the proposed training extension can help boost the ROUGE scores of any summary generators.

1 INTRODUCTION

Text summarization is a Natural Language Processing task in which a system is designed to summarize a text or document in a much shorter number of words, compressing the information contained so that it is both understandable by humans and still contains the core themes of the original document. This is a relevant topic of research as it can be applied directly to enable humans to be more efficient and understand the core of a document without needing to read its entirety. Text summarization can also be used to plug into existing information retrieval pipelines, for instance in order to simplify a long query into a more concise format, or compress the information in documents to speed up searching through a large collection.

Text summarization is split into two categories: extractive and abstractive. The first deals with extracting the summary directly from the text by selecting passages of it and piecing them together. The other approach, and the one that this research focuses on, attempts to summarize texts using abstractions and allows for outside vocabulary to be used. This is more aligned with how humans approach text summarization, and forces the models to abstract information rather than just selecting it.

Training abstractive text summarization models remains a challenging task as it involves extracting key concepts and abstractions from a text and compressing them into a readable format. In general, neural abstractive summarization models used are trained by maximizing the likelihood of the reference summary. Since this might lead to low-quality generations or even incorrect sentences, Piji et al proposed the use of an actor-critic approach [6], in which one of the proposed critic networks is a binary classifier network similar to the discriminator in a Generative Adversarial Network. The critic network takes in generated summaries and reference summaries and attempts to tell which are fake. This forces the architecture during training to generate summaries which are indistinguishable from human summaries and has been shown to achieve improvements over state of the art results [6].

However, a limitation of this training technique is that it bases critic loss on the final generated target, and so serves only as a quality estimator of fully generated summaries. Piji et al use the REINFORCE algorithm with an alternating training strategy to train their critic network [6]. The use of their critic network to include an adversarial loss strategy also brings difficulty in training time and many tricks have to be used in order to obtain the networks to even converge. Also, their critic is completely global level with no emphasis on the word level qualities.

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [7] is the metric of choice when evaluating text summarization, and qualitatively compares generated summaries to human reference summaries. In this paper, we propose using the ROUGE metric as an explicit *local* (word level) and *global* (sentence level) quality estimator for abstractive text summarization, re-weighting the loss of each sub-sequence by the ROUGE score difference they generate.

This change allows our loss to be word or sentence based rather than summary based and thus better model the target summary. Rather than averaging the loss over the summary, words which contribute a lot or very little to the total ROUGE score are learned through this model, and the model is thus forced to learn summarization as a more abstractive task versus an extractive task. We also experiment with mixing *local* and *global* training approaches to provide the generator with a wholesome quality evaluator. Learning using ROUGE also allows us to optimize directly for it efficiently, and thus obtain better results than the standard baseline approach.

Conference'17, July 2017, Washington, DC, USA
2019. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

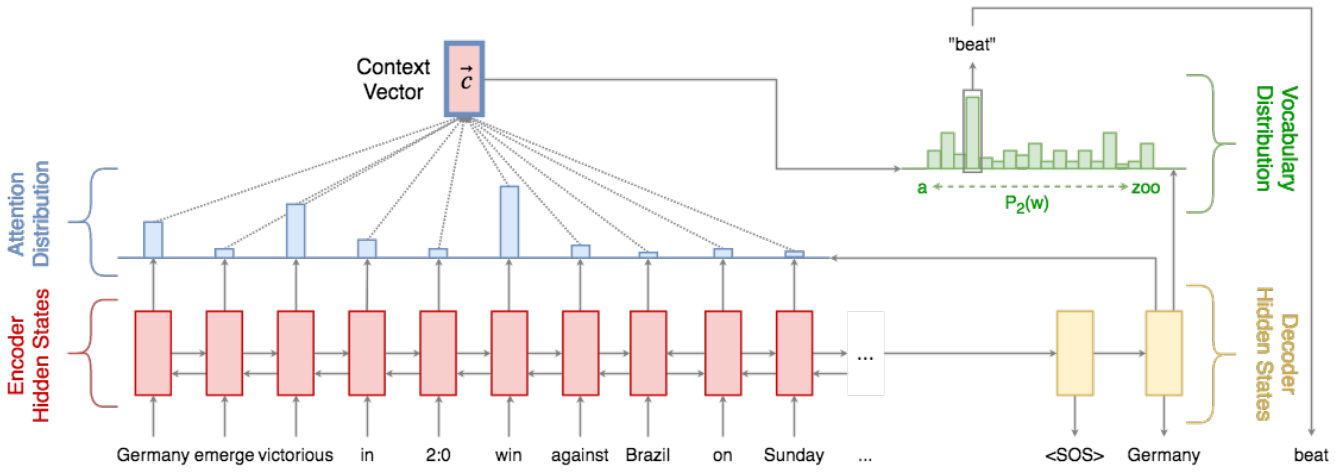


Figure 1: Model Architecture

2 RELATED WORK

Due to the difficult nature of generating abstractive sequences, most work in text summarization has historically been focused on extractive methods [11]. However, the advent of Sequence to Sequence (Seq-2-Seq) models, relying on Recurrent Neural Networks, has opened up more research into the generation of viable abstractive summaries.

Nallapati proposed a Seq-2-Seq model architecture using attention for encoding a text and decoding it into a summary in 2016 [10]. See et al. [12] then proposed the Pointer-Generator network for text summarization in 2017, which combined Nallapati's approach and the Pointer Networks architecture proposed by Vinials [13]. See's Pointer Network architecture allows both copying words via pointing, and generating words from a fixed vocabulary [12]. This architecture is therefore able to reach a higher ROUGE score, by abstracting general phrasing but still retaining document specific information in the text and thus mimicking the human-made references. See et al. also introduced the coverage loss mechanism to force the attention to areas of the sentences with least likelihood. This innovation forces the Pointer-Generator to focus on different parts of the sentence and thus limit repetition in the generated summary.

Hsu et. al. [5] further improved on this Pointer Generator model by introducing an inconsistency loss which combines the previously used word level attention with a sentence level attention over the entire document. Instead of limiting the summarization to the first 20 sentences as in See et al. [12], the entire document can be incorporated using this mechanism, resulting in significant improvements. Similarly Gerhmann et al. [3] used a content selection step on the document sentences to then mask the irrelevant sentences and pass the result through a Pointer-Generator network. They obtain similar improvements as [5], and also have the advantage of not having the input be length restricted.

Another improvement was made by Xiang et al. [14] on the Pointer-Generator network by measuring the cosine similarity of an encoded reference summary with the encoded document and using it as a loss component. This forces the Pointer-Generator network to encode the most relevant information to summaries and thus obtain better ROUGE scores.

Since these approaches are Maximum Likelihood Estimation (MLE) based models, they suffer from the optimization of the KL-divergence in a single direction. Changing the loss allows us to move away from problems caused by using an MLE estimation and design a more natural abstraction quality which our neural network can learn.

This has been addressed in the past by applying Generative Adversarial Networks (GANs) to the sequence generation task. GANs inherently solve the KL-divergence optimization imbalance through the use of a discriminator, whereas the introduction of Policy Gradients in the loss functions of the GANs help to eliminate exposure bias. Lui et. al. apply the GAN architecture to Pointer Generator networks using these Policy Gradient updates and demonstrate that they achieve competitive results [8]. They use the Pointer Generator network as the generative portion of the GAN network, but in order to discriminate properly between sample data and real summaries, the discriminator is only able to classify on the final generated summaries.

This issue with using this GAN approach is the fact that discriminators still only classify the completed sequences. Some other GAN approaches have addressed this issue in other domains such as the SeqGAN network which applies Monte Carlo Tree Search in order to estimate the final outputs of the discriminator given an intermediate state in the sequence generation [15]. This enables the intermediate Policy Gradients to be more accurately weighted by their contributions to the final loss and thus offer an intuitive way to evaluate sub-sequences.

Yet SeqGAN is very computationally expensive and there are simpler ways to proxy a local quality estimator for summarized subsequences, namely to use the evaluation metric ROUGE directly. This paper implements using a ROUGE based quality evaluator as a direct Critic Policy in order to evaluate generated summaries both locally (word-level) and globally (sentence-level).

3 METHOD

3.1 Model

Our model is a sequence to sequence attention neural network structured similarly to the baseline model proposed in See et al [12] and is shown in figure 1. The inputs to the model are the first 20 sentences of a given source document or article. This is done in order to speed up training time and convergence and compare results to the See et al. paper since it is also done there [12].

3.1.1 Architecture. Each token w_i from this shorter document segment is passed into a single Bidirectional LSTM layer to obtain a set of encoded hidden states $\boldsymbol{h}$. Then at each generation time step t, the previously generated token $\hat{w}_t$ (during training this is the previous token in the reference summary) is passed as a word embedding into a uni directional decoder LSTM layer. We then use the output of the decoder s_t and the hidden states of the encoder $\boldsymbol{h}$ in a simple attention layer as proposed by Bahdanau et al. [1], where the attention distribution at time step t is $\boldsymbol{a_t}$:

$$e_t^i = v^\top \tanh(W_h h^i + W_s \boldsymbol{s_t} + b_a) \tag{1}$$

$$\boldsymbol{a_t} = \text{softmax}(\boldsymbol{e_t}) \tag{2}$$

We then use the attention distribution to obtain the context vector $\boldsymbol{c_t}$ which is a weighted sum of the original hidden encoder output:

$$\boldsymbol{c_t} = \sum_{i=1} a_t^i h^i \tag{3}$$

The context vector is then concatenated with the decoder output state and passed through a two layer feed-forward network which outputs a probability distribution over the vocabulary set $p_{\text{vocabulary}}$:

$$p_{\text{vocabulary}} = \text{softmax}(B(A[\boldsymbol{s_t}, \boldsymbol{c_t}] + b_A) + b_B) \tag{4}$$

Where A, B, b_A, b_B are learnable parameters denoting the weights and biases of the two layers. To then select the next word we sample from this distribution using Beam Decoding with $n = 4$ and keep the best sequence.

3.1.2 Loss. Initial training is done using the standard Maximum Likelihood approach, or using the Negative Log Likelihood (NLL) to calculate our loss with respect to the target word w_t^*:

$$G_\theta(w_t) = -\log p_{\text{vocabulary}}(w_t^*) \tag{5}$$

This is the baseline loss used in the baseline by Lee et al. [12], and is used here in order to initialize weights and warm up the model before the new training schedule. Due to the simplicity of the loss calculations, we noticed that our models were able to converge faster initially using this warm up stage.

Inspired by the Policy Gradient Loss as used in the SeqGAN, we then improve on the NLL loss by weighing the loss of the sequence by the individual rewards (state-action values) of the individual tokens for the entire sequence [15] (This is referred to as local reward policy learning in later sections):

$$\nabla_\theta J(\theta) = \sum_{t=1}^{T} \nabla_\theta G_\theta(w_t) Q(w_t, w_1^{t-1}) \tag{6}$$

This loss can also be divided up at a sentence level for each generated summary as follows:

$$\nabla_\theta J(\theta) = \sum_{i=1}^{|S|} \sum_{j=1}^{|S_i|} \nabla_\theta G_\theta(w_{i,j}) Q(w_{i,j}, w_{i,1}^{i,j-1}) \tag{7}$$

where $w_{i,j}$ denotes the j-th word in the i-th sentence.

The reward function Q that is used is a modified version of the ROUGE metric. In order to more accurately weigh the contributions of the individual words and sentences to the sequence so far, a percentual ROUGE weighting Q_{weight} is used, which can be used with any type of ROUGE sub-metric, for example ROUGE-S:

$$Q_{weight,S}(S, S_{-i}) = \frac{\text{ROUGE-S}(S) - \text{ROUGE-S}(S_{-i})}{\text{ROUGE-S}(S)} \tag{8}$$

Where S is the set of all sentences in the summaries, and S_{-i} denotes the same set excluding the i-th sentence.

Therefore, the individual token level rewards can also be replaced with a sentence level reward (It is referred to as global reward policy learning in future sections):

$$\nabla_\theta J(\theta) = \sum_{i=1}^{|S|} \sum_{j=1}^{|S_i|} \nabla_\theta G_\theta(w_{i,j}) Q(S, S_{-i}) \tag{9}$$

Naturally it follows that these can also be combined in a single unified function, utilizing both sentence and word level reward functions:

$$\nabla_\theta J(\theta) = \sum_{i=1}^{|S|} \sum_{j=1}^{|S_i|} \nabla_\theta G_\theta(w_{i,j}) (\alpha Q(w_{i,j}, w_{i,1}^{i,j-1}) + (1-\alpha) Q(S, S_{-i})) \tag{10}$$

where α is a weighting hyper-parameter between the levels. We refer to this as mixed reward based policy learning.

4 EXPERIMENTAL SETUP

4.1 Dataset

Similar to See et al [12], we use the **CNN/Daily Mail DeepMind Q & A dataset** [4], which includes online news articles with questions, answers and multi-sentence reference summaries. We use only the stories (781 tokens on average) from this dataset, which has around 92,579 CNN stories and 197k DailyMail stories. The dataset is split as around 267k training pairs, 13k validation pairs and 11k testing pairs. Each pair consists of a story and corresponding reference summary (3.75 sentences and 56 tokens on average). Similar to the work of See et al [12], we directly use the original, non-anonymized stories/summaries. Further these stories are tokenized using *Stanford CoreNLP* tokenizer [9]. Finally, these tokenized stories are lower-cased and converted as serialized binary files. In order to achieve this, we used the work of See et al [12]. [1]

4.2 Evaluation: ROUGE

Throughout this work, ROUGE [7] score is used as the evaluation metric for the summarization work. It measures the quality of the generated summary to the reference summary, typically written by humans. ROUGE has mainly 5 different evaluation metrics of

[1] https://github.com/abisee/cnn-dailymail.git

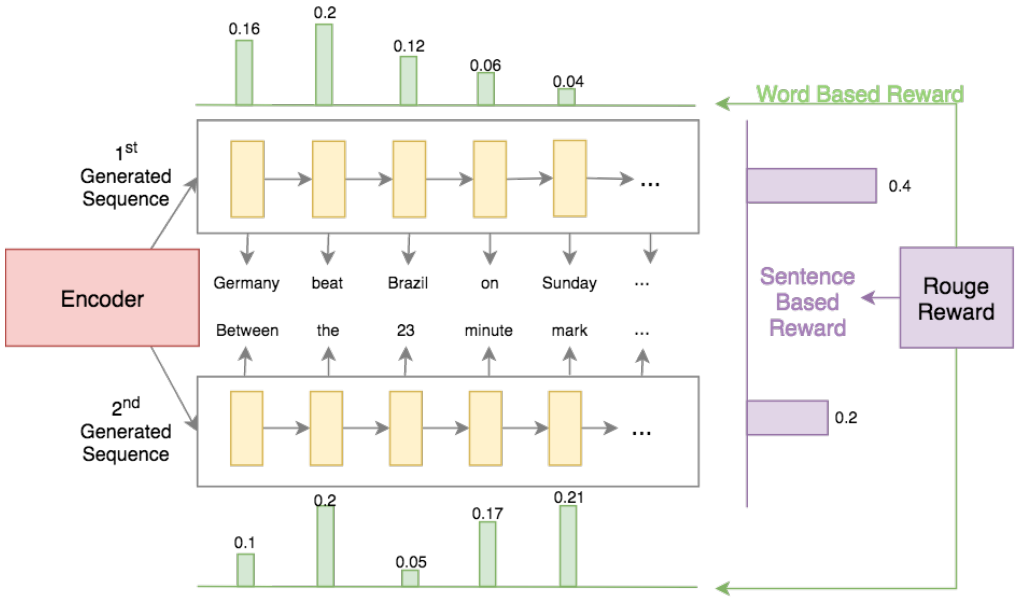


Figure 2: Reward Procedure

which ROUGE-N and ROUGE-L are used in this work. ROUGE-N measures the overlapping of N-grams between the generated summary and reference summary. We use ROUGE-1 (unigram or single word) scores for word level evaluation in our experiments. ROUGE-L measures the sentence level structure similarity between the generated and reference summary, using Longest Common Subsequence (LCS)[7] based statistics. We use this for sentence level evaluation of the generated summaries.

Although we use the official Perl based rouge for the final evaluation, we use the python based rouge evaluator [2] for getting the rouge scores during training. This is done mainly for a faster training. For training, we use the F-1 scores given by the rouge evaluator to compute the rewards. All the testing results also show the F-1 scores for the specified model.

All three reward methods (local, global, and mixed) are implemented in code in the seqGan/rewards.py file.

4.3 Hyper-parameter settings

Similar to the work of See et al, we use 256 dimensional hidden state and 128 dimensional word embeddings. Since there is no change in the model from See et al baseline model, there are 21.5 million parameters. Also, the word-embeddings are not pre-trained but instead learned from scratch during the training process.

Similar to the work of See et al, we use a vocabulary of 50k words for both source and target since it is much faster due to the small size and gives slightly better results. We also tried various vocabulary sizes of 25k, 50k, 75k, 100k and 200k on the baseline of which 50k has the best results and faster (except 25k for obvious reasons) compared to other vocabulary sizes. This could be because of the increased word options that might get misused due to a larger distribution size.

[2]https://github.com/pltrdy/rouge

For training we use Adagrad optimizer [2] with learning rate of 0.5 and initial accumulator value of 0.1. We also tried using Adam and RMSProp optimizers which resulted in lower quality summaries. Further, a gradient clipping with maximum gradient norm of 2.0 is used. We do not use any other forms of regularization techniques. For the mixed reward strategy, we use an alpha value of 0.3. During training, we use the word with maximum probability distribution for generating the summary. During testing, we used beam search with beam size of 4 for generating best summaries.

We trained both the Pointer-Generator and Sequence-to-Sequence baseline models for 50k iterations (each iteration corresponds to the training of one batch), which is a bit more than 3 epochs. The reward based policy learning training and baseline approach (standard MLE) is done for 8000 iterations using the trained (50k iterations) Seq-to-Seq model baseline checkpoint. See Appendix A for replication of results from See et. al. See et al. [12], where we justify the selected checkpoint. This is equivalent to training the model for about 4 epochs considering the 3 epoch pre-trained baseline checkpoint.

5 RESULTS AND ANALYSIS

5.1 Proposed Methods

As explained in Section 3, the proposed model uses the rouge scores to compute the rewards and thus the loss. Based on the way of calculating these rewards, we have three main types: local, global, mixed. All our experiments regarding reward based strategies are done on the Sequence-to-Sequence model proposed in section 3.1, and only the loss is changed during training according to which reward approach we are testing.

Note that we believe using the same loss on Pointer-Gen would improve results further, but chose to prove our approaches on Seq-to-Seq due to time and computational limitations.

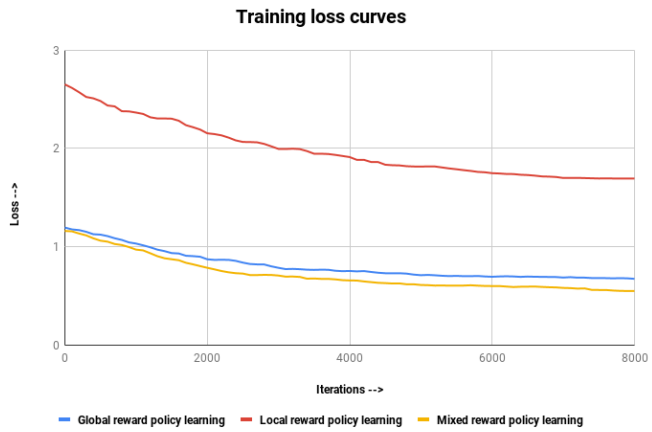


Figure 3: Training Loss curves of reward based policy learning

	ROUGE-1	ROUGE-2	ROUGE-L
Baseline (Seq-Seq, 58k)	24.39	7.33	21.97
Global reward	26.98	8.99	26.81
Local reward	29.88	10.95	24.38
Mixed reward	**29.93**	**11.04**	**27.48**

Table 1: Seq-to-Seq ROUGE scores of Baselines and Proposed Training Approaches

5.1.1 Global reward strategy. The global reward strategy uses the sentence level rewards as explained in equation 9. For this, we train the model by using the ROUGE-L f1 score in order to reweigh the loss of the generated sentences with respect to how much of the total ROUGE score they generated. The results are shown above in table 1 where this method obtained a score of 26.98 on ROUGE-1, 8.99 on ROUGE-2 and 26.81 on ROUGE-L. Clearly, this is an improvement over the baseline model. Figure 4 shows how the rouge scores, especially ROUGE-L, improves through the iterations. We can observe that Global reward strategy has better ROUGE-L score compared to local reward strategy as it mainly focuses on improving the generated sentence quality. This can be observed consistently during the training.

5.1.2 Local reward strategy. The local reward strategy uses the word level rewards and computes the loss as given by equation 6. In this method, we train the model by using the ROUGE-1 f1 score in order to reweigh the loss of the generated sentences with respect to how much of the total ROUGE score they generated. The results are shown above in table 1 where this method obtained a score of 29.88 on ROUGE-1, 10.95 on ROUGE-2 and 24.38 on ROUGE-L. This is already an improvement on the baseline and global approach for ROUGE-1 and ROUGE-2. However, the global approach still has a better ROUGE-L score than the local approach because that is what it optimizes for directly.

	ROUGE-1	ROUGE-2	ROUGE-L
$\alpha = 0.1$	27.49	9.86	27.13
$\alpha = 0.3$	29.06	10.14	27.21
$\alpha = 0.5$	29.84	10.71	27.35
$\alpha = 0.7$	**29.93**	**11.04**	**27.48**
$\alpha = 0.9$	29.92	10.99	26.11

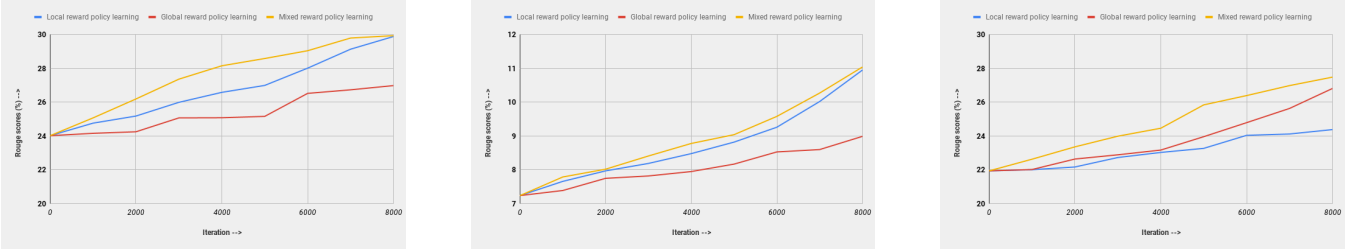
Table 2: Effect of mixing hyper-parameter α on mixed reward policy learning strategy

5.1.3 Mixed rewards strategy. The mixed reward strategy uses both word and sentence level rewards as given by equation 10. It uses an extra tunable hyper-parameter α to control for which reward to use more in reweighing. The optimal $\alpha = 0.7$ was found after testing multiple values. The mixed strategy results are also shown above in table 1 where this method obtained a score of 29.93 on ROUGE-1, 11.04 on ROUGE-2 and 27.48 on ROUGE-L. As we can see this method surpasses all the others since it optimizes on both the ROUGE-1 and and ROUGE-L metrics simultaneously. It is interesting to note that it surpasses even the ROUGE-1 of the local approach and the ROUGE-L score of the global approach. Figure 4 clearly demonstrates this. This could be due to the fact that the mixed approach offers a more complete evaluation of the different quality of a generated summary and as a result decreasing over-fitting any particular type of rouge metric, while increasing abstractiveness.

Table 2 demonstrates the effect of mixing parameter α in the mixed reward strategy. When the α value is small (ie close to 0), the weight is given more on the sentence level rewards pushing the model to get tuned more on ROUGE-L based rewards. On the other hand when the α value is high (ie close to 1), the weight is given more on the word level rewards leading to lower ROUGE-L scores. This makes sense, as the smaller α value gives more weight on the global rewards and thus giving a higher ROUGE-L score but lower ROUGE-N scores and the larger α value gives more weight on the local rewards thus resulting in higher ROUGE-N scores but lower ROUGE-L scores. An optimal value of $\alpha = 0.7$, optimally balances the global and local rewards resulting in best ROUGE scores on both word and sentence level, giving slightly more weight on word level rewards.

5.2 Case studies

In Table 3 are shown example generated summaries for each generator trained on all 4 different approaches respectively and stopped at the same iteration point. We note immediately the diversity in generated summaries, even though all generators were initialized with the same weights. Errors are colored in red, novel words in green, and satisfactory rephrasing in orange. Repeating words/phrases in the summary are denoted in blue. Directly extracted phrases from the original article are highlighted in yellow.



R-1 scores **R-2 scores** **R-L scores**

Figure 4: ROUGE score evaluation of the proposed methods during the training process. The Local rewards based method (Blue) has better R-1 and R-2 scores compared to Global rewards based method (Red). However, Global rewards based method (Red) has better in R-L score compared to Local rewards based method (Blue). Mixed reward based method (Yellow) has the best rouge scores in all categories.

We find the baseline to be very limited in its ability to summarize, similarly as was found by See et. al. [12] in the Get to the Point paper. We can observe several direct references to the original article, such as *the video was found by a source on board flight*. Also, the baseline has several repeating words and phrases similar to the baseline of See et al. The baseline also has few factual errors such as using the wrong statement headline *new*. It also suffers from OOV words and replaces them with [UNK]. This can be clearly observed in the phrase *on board flight [UNK] flight [UNK]*, whereas the same phrase in original article is *on board flight Germanwings Flight 9525*. Another drawback we observe in baseline is severe generation of nonsensical sentences such as *it's [UNK] in the world* in the example. The sentence as such makes no sense and the words are taken randomly from all over the article.

Global rewards seem to slightly improve the diversity of sentences compared to the baseline. Although it uses the wrong statement head *new* first, it immediately follows the correct statement head *French prosecutor*. We can also observe the reduction of OOV word. In addition to that, the model generates the novel words such as *expert* and *analyst* which are not present anywhere in the original article. Thus using a global reward strategy helps getting rid of some of the issues in the baseline model. However, it still suffers from the repeating words/phrases and direct references to the article, which can be clearly seen in Table 3.

Local rewards work better than the baseline, suffering less from OOV words and nonsensical sentences. We can observe the sentence *the video was found on board flight Germanwings Flight 9525*, which consists of the OOV words, is smartly rephrased as *the video was found at site*. Also, we can observe it replaced the *Germanwings Flight 9525* with a simple word *plane*, and thus not using the [UNK] words. This is probably because the model focuses on word level quality of the summary and thus encouraging it to rephrase the sentence with similar/abstract meaning. However, it still suffers from the repeating words/phrases and direct reference to the original article.

Mixed rewards fared much better, showing improved rephrasing qualities as well as novel word generation in the summaries. Since it is a combination of local and global rewards, it cherishes the advantages of both. We can observe the usage of good rephrasing of the sentence *previous episode of severe depression* as *about previous depression*. Also, we can observe the usage of novel words such as *expert* and *claims*, which are not present in the original article. We also observe the reduction in the usage of statement head *new* in mixed reward strategy. We also observe lesser direct referencing in this strategy. These are all signs of learned abstractiveness by the generator and justifies the larger ROUGE scores obtained through this approach. However, it still suffer from the repeating words/phrases. As per See et al [12], this can be overcome with the usage of coverage mechanism.

More similar case studies are presented in appendix. In all the cases we can observe baseline has lots of grammar and factual errors. The global and local reward try to fix this by using more direct references to the original article. They also suffer from repeating words/phrases. The mixed strategy, our best proposal, has least errors and very less repeating words/phrases with small amount of novel words usage and good rephrasing. However, it suffers from sever direct referencing. As per the work of See et al [12], the direct referencing can be reduced using pointer generator networks and repeating words/phrases can be avoided with the usage of coverage mechanism.

6 CONCLUSION

We found through our experiments that optimizing on ROUGE directly during training does translate to higher results in evaluation. The most successful approach being the mixed approach with which we are able to considerably surpass a baseline generator trained solely using MLE.

We show that both the local and global rewards also led to ROUGE improvements over the baseline, and that unsurprisingly, optimizing a specific ROUGE metric directly and positively impacts the score of that metric during evaluation. Additionally, we found that using a mixed approach surpassed even the local ROUGE-N

Article: (parts)
Marseille, France (CNN)The French prosecutor leading an investigation into the crash of Germanwings Flight 9525 insisted Wednesday that he was not aware of any video footage from on board the plane. (...) a cell phone video showing the harrowing final seconds from on board Germanwings Flight 9525 (...) Marseille prosecutor Brice Robin told CNN that "so far no videos were used in the crash investigation." (...) " It is a very disturbing scene ," said Julian Reichelt, editor-in-chief of Bild online. (...) Paris Match and Bild reported that the video was recovered from a phone at the wreckage site. (...) Lubitz told his Lufthansa flight training school in 2009 that he had a "previous episode of severe depression," the airline said Tuesday. (...)
Reference Summary:
marseille prosecutor says " so far no videos were used in the crash investigation " despite media reports . journalists at bild and paris match are " very confident " the video clip is real , an editor says . andreas lubitz had informed his lufthansa training school of an episode of severe depression , airline says .
Baseline (Seq-Seq) :
new : " it is a very hate scene , " the company says .
the video was found by a source on board flight [UNK] flight [UNK]
the video was found by a source on board flight [UNK] flight .
he says " it 's [UNK] " in the world .
Global rewards :
new : the french prosecutor : ' this is a very disturbing scene , ' says police .
the video was recovered from a phone at the wreckage site .
he said : ' it is a very disturbing scene , ' says expert .
' ' it is a very disturbing scene , ' ' says analyst .
Local rewards :
a man says he was not aware of any video footage from on board the plane.
he said he was not aware of any video footage from on board the plane.
the video was found at the site .
Mixed rewards :
french prosecutor says he was not aware of any video footage from on board the plane .
the expert claimed the video clip was found by a source close to the investigation .
the airline said the pilot informed the training school about previous depression .

Table 3: Comparison of summaries on a CNN article. (Color map: Error, Novel Words, Abstract Rephrasing, Repetition, Direct Extraction ,)

and global ROUGE-L scores, which indicates that our abstractiveness quality estimator might be better represented as a combination of ROUGE metrics rather than a single one.

Through analysis we show that ROUGE score differences between our approach and the baseline approach are significant enough to make a difference at an understandable human level. Therefore, we expect that using the proposed training extension can help boost the ROUGE scores of any summary generators it is applied to. Since our training extension does not require much tuning (a single α hyper-parameter), using it in current state-of-the-art summary generators could potentially quickly lead to increasing abstractiveness and performance.

There are however possible limitations to optimizing training using ROUGE reweighing. Reweighing loss using a distribution decreases some of the gradients pushed back to the network during back-propagation to focus more attention on words or sentences which led to the least increase in ROUGE score. We therefore expect learning to be slower than the traditional MLE training since losses are multiplied with a function of the percentage gain they provide $Q \in (0, 1)$. Such a training procedure might then require a longer training period or adding an additional hyper-parameter to boost the losses in order to increase the gradients pushed to the back-propagation step.

Additionally, we also found that the CNN/Dailymail dataset has important limitations for evaluating accurately the abstractive properties of our generator. Since every article only contains a single annotated "reference" summary which the ROUGE depends on, the model is trained to emulate the single reference summaries. This makes the ROUGE score very coarse and unable to capture possible alternate ways of summarizing the same text. Additionally news articles are very structured texts and their reference summaries often contain pieces of facts directly extracted from the text which makes it poorly suited for evaluating abstractive summarization.

While our approach sought out to implement a local Critic Policy, it remains to be seen whether using a learnable Critic Policy on words and sentences of the summary also improve the abstractions of the generator. This could be done by implementing a SeqGAN with our generator network which remains as a potential avenue for future work with more available computational resources. Furthermore, it would be interesting to test the Pointer Generator architecture as the generator network instead of using our Sequence-to-Sequence architecture, but again this would require additional resources.

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR* abs/1409.0473 (2014). arXiv:1409.0473 http://arxiv.org/abs/1409.0473

[2] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research* 12, Jul (2011), 2121–2159.

[3] Sebastian Gehrmann, Yuntian Deng, and Alexander M. Rush. 2018. Bottom-Up Abstractive Summarization. *CoRR* abs/1808.10792 (2018).

[4] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*. 1693–1701.

[5] Wan Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. 2018. A Unified Model for Extractive and Abstractive Summarization using Inconsistency Loss. In *ACL*.

[6] Piji Li, Lidong Bing, and Wai Lam. 2018. Actor-Critic based Training Framework for Abstractive Summarization. *CoRR* abs/1803.11070 (2018). arXiv:1803.11070 http://arxiv.org/abs/1803.11070

[7] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out* (2004).

[8] Linqing Liu, Yao Lu, Min Yang, Qiang Qu, Jia Zhu, and Hongyan Li. 2018. Generative Adversarial Network for Abstractive Text Summarization. *CoRR* abs/1711.09357 (2018).

[9] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*. 55–60.

[10] Ramesh Nallapati, Bing Xiang, and Bowen Zhou. 2016. Sequence-to-Sequence RNNs for Text Summarization. *CoRR* abs/1602.06023 (2016). arXiv:1602.06023 http://arxiv.org/abs/1602.06023

[11] Horacio Saggion and Thierry Poibeau. 2013. Automatic text summarization: Past, present and future. In *Multi-source, multilingual information extraction and summarization*. Springer, 3–21.
[12] Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. In *ACL*.
[13] O. Vinyals, M. Fortunato, and N. Jaitly. 2015. Pointer Networks. *ArXiv e-prints* (June 2015). arXiv:stat.ML/1506.03134
[14] Xiujuan Xiang, Guangluan Xu, Xingyu Fu, Yang Wei, Li Jin, and Lei Wang. 2018. Skeleton to Abstraction: An Attentive Information Extraction Schema for Enhancing the Saliency of Text Summarization. *Information* 9, 9 (2018). https://doi.org/10.3390/info9090217
[15] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2016. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. *CoRR* abs/1609.05473 (2016). arXiv:1609.05473 http://arxiv.org/abs/1609.05473

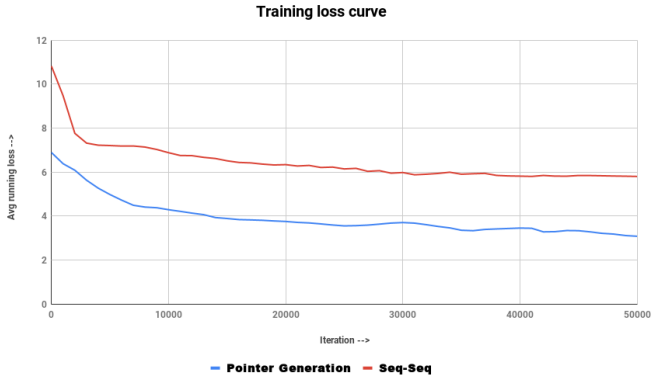


Figure 5: Training Loss of Pointer Generation and Seq-Seq models

	ROUGE-1	ROUGE-2	ROUGE-L
Pointer-Gen (500k)	36.44	15.66	33.42
Pointer-Gen (50k)	34.18	14.08	30.67
Seq-Seq (500k)	31.33	11.81	28.83
Seq-Seq (50k)	24.02	7.24	21.94

Table 4: ROUGE Scores of Replicated Baselines

A APPENDIX: POINTER-GENERATOR VS SEQ-TO-SEQ

Seeking to reproduce the original results obtained by See et. al. [12], we trained the original pointer-generator and Seq-to-Seq baselines from the 2017 paper for 50k iterations. We report our results in Table 4, and Figure 5. We observe that both the pointer-generator and baseline Seq-to-Seq models converge quickly and obtain satisfying results for training only a tenth of the original iterations. The original results were trained for 500k iterations which only marginally improved their ROUGE score and are also reported in Table 4. Due to time and computational limitations, a Seq-to-Seq checkpoint at 50k iterations was deemed a satisfactory starting point upon which to append our reward based training approaches.

B APPENDIX: ADDITIONAL EXAMPLES

This appendix provides some examples from the test set. In each example, the original article, reference/human summary, and generated summaries from baseline and our methods are provided. In all of the examples, we use following color mapping for ease of understanding.

- Red : Denotes the errors, mistakes, wrong facts and [UNK] characters.
- Green : Denotes the usage of Novel words. These words are not present in the original article but present in the vocabulary.
- Blue : Denotes repetition of the words or sentences.
- Orange : Denotes a good rephrasing of the sentence or summary.
- Yellow : Denotes the direct reference of few words from the original article.

Article:
(CNN) Paul Walker is hardly the first actor to die during a production . (...) But Walker's death in November 2013 at the age of 40 after a car crash (...) The actor was on break from filming "Furious 7" at the time of the fiery accident, which also claimed the life of the car's driver , Roger Rodas. (...) Vin Diesel gave a tearful speech before the screening, saying "This movie is more than a movie." (...) I know that we will never forget him and he will always be someone very special to us ," said Upham. (...) The release of "Furious 7" on Friday offers (...)
Reference Summary:
" furious 7 " pays tribute to star paul walker , who died during filming . vin diesel : " this movie is more than a movie " " furious 7 " opens friday .
Baseline (Seq-Seq) :
new : " i 'm going to get out , " he says . new : " it 's not a good thing , " he says . [UNK] says it 's not to be a good story of the film .
Local rewards :
paul walker is the first actor to die during a production . he was the first actor to die at the time of the car crash . he is now the first actor to die .
Global rewards :
paul walker 's death in november 2013 at the age of 40 after a car crash was key . the actor was on break from filming ' furious 7 ' at the time of the crash accident , which also claimed the life of the car 's driver , roger dickson .
Mixed rewards :
paul walker 's death in november 2013 at the age of 40 after a car crash . the actor was on break from filming " furious 7 " at the time of the fiery accident . i know that we will never forget him and he will always be someone very special to us .

Table 5: Example-1: This is very challenging article, with very human-like reference summary. The baseline fails miserably by making nonsense statements, although grammatically they are somewhat correct. The local rewards is better than the baseline. However, it misses the word *hardly* from the original sentence ...*hardly the first actor*... and makes factual errors with sentences such as ... *the first actor to die*. The global rewards get's rid of these smart mistakes, but ends up with very high direct references. We can also observe the wrong name of the driver as *Roger Dickson*, whereas originally it is *Roger Rodas*. The mixed rewards has relatively less errors, but has very high direct references. This is one corner case where the reference summary has direct reference to statements in the article.

Article: (parts)
(CNN)Seventy years ago, Anne Frank died of typhus in a Nazi concentration camp at the age of 15 . (...) But new research released by the Anne Frank House shows that Anne and her older sister , Margot Frank, died at least a month earlier than previously thought . (...) That same year, Anne and Margot were separated from their mother and sent away to work as slave labor at the Bergen-Belsen camp in Germany. (...) hey concluded that Anne and Margot probably did not survive to March 1945 (...) Her diary endures as one of the world's most popular books.
Reference Summary:
museum : anne frank died earlier than previously believed . researchers re-examined archives and testimonies of survivors . anne and older sister margot frank are believed to have died in february 1945 .
Baseline (Seq-Seq) :
new : the [UNK] of the [UNK] was found dead in [UNK] , new york . she was found dead in the [UNK] in [UNK] , texas . she was found dead in the [UNK] in [UNK] , texas . she was found dead in the [UNK] in [UNK] , texas .
Global rewards :
anne frank died at least a month earlier than previously thought . she died of [UNK] in a nazi concentration camp at the age of 15 . she was found dead in the [UNK] concentration camp in germany at the age of 15 .
Local rewards :
new research released by the anne frank house that anne and her older sister , cindy frank , died at least a month earlier than previously thought . new research released by the anne frank house that anne and her older sister , cindy frank , died at least a month earlier than previously thought he was liberated .
Mixed rewards :
anne frank died of [UNK] in a nazi concentration camp in germany at the age of 15 . she died at least a month earlier than previously thought . the pair were separated from their mother and sent away to work as slaves . new research released by the anne frank house shows that anne and her older sister , cindy frank , died at least a month earlier than previously thought .

Table 6: Example-2 : In this example the baseline fails miserably with lots of repeating, direct referencing statements along with several factual errors. The global rewards reduces the factual errors and repetitions, but still has several direct references. The local rewards again repeats the sentences. It also makes a factual mistake of anna franks elder sisters name as *Cindy Frank* whereas it should be *Margot Frank*. Mixed rewards strategy has better summary, but still has several direct references. It iasl retains the factual error of *Cindy Frank*. However, as a summary it is grammatically quite right and has the gist of the article.

Article: (parts)
(DailyMail) In a Versace gown at this year's Oscars, Jennifer Aniston showed off her well-toned arms . The 46-year-old works out three times a week with her long-term trainer , Mandy Ingber. (...) Sessions include 20-25 minutes of spinning, followed by 45 minutes of yoga . (...) When she stays at a hotel, Aniston takes 3.6kg weights with her. She says exercise, plus meditation, is her 'ultimate stress reliever' . (...) The medicine ball overhead press is great for upper arms. Ideally, use a weighted medicine ball, but a football is fine for beginners. (...)
Reference Summary:
jennifer aniston , 46 , showed off her well-toned arms at the oscars .
she works out three times a week with her long-term trainer , mandy ingber .
what to try : the medicine ball overhead press is great for upper arms .
Baseline (Seq-Seq) :
the duchess of the duchess of cambridge is the most popular in the world .
the duchess of the duchess of cambridge is the first of the year .
the duchess of the duchess of cambridge is the first of the year .
Global rewards :
jennifer aniston 's arms is the first time in a glamour gown at this year 's oscars .
the pair were inspired by 45 minutes of yoga .
when she moved at a hotel , pumpkin takes off her [UNK] arms .
Local rewards :
jennifer is the first person of the year 's oscars .
she says exercise , plus meditation , is her ' ultimate stress healer ' she says exercise plus, medication , is her ' ultimate stress d' '
Mixed rewards :
jennifer aniston takes 3.6 kg weights with her .
she works out three times a week with her long-term trainer , mandy [UNK] .
25 minutes of spinning , followed by 45 minutes of yoga .

Table 7: Example-3 :In this example the baseline fails badly with several repeating wrong comments with many grammar mistakes. The global rewards is relatively better with few factual errors. it also generates a novel word *pumpkin*, which is a grammatical error in the summary. It also has several direct references. The local rewards has less errors, but has several grammar mistakes. It also has several direct references along with repeating words/phrases. The mixed strategy has less errors and mistakes. It has no repeating words/phrases. However, we can observe several direct references to the original article.