Improving ROUGE for Abstractive Summarization

Eshed Gal
Tel Aviv University
eshedgal@gmail.com

Ohad Yousfan
Tel Aviv University
ohadyousfan@gmail.com

Abstract

Language models today can do different complex tasks, one of which is text-summarization. In order to evaluate the quality of the generated summary, different metrics are used that try to quantify syntactic and semantic properties of the summary and relate it to a reference summary and/or to the original text. A popular metric to do so is ROUGE, that compares overlaps of n-grams in the model generated summary to a reference summary. As human summaries are complex and expensive, the goal is to create models that can generate summaries that are as good as human summaries. An integrated part of that is the need to automate the ability to qualitatively evaluate the summary, and to do so as efficiently as possible. Despite it's popularity, ROUGE has several flaws, such as giving a high score to summaries that are not factual or not coherent. We aim to improve ROUGE by mitigating some of it's flaws while still keeping it's qualities. We examine regularization factors that refer to the original text and to the reference summary in a semantic and extractive manner to adjust the final ROUGE score. We show that we get better results compared to the classic ROUGE metric.

1 Introduction

A summary of a text is a text of shorter-length that contains the primary ideas of the original text in a concise and a coherent manner. A good summary should contain all important points from the original text, and should not contain redundant information, or information that is not present in the text. There are different types of summaries. A summary is affected by the length of the text, the domain and writing-style, the summary length and its goal. Two major types of summaries are extractive, in which there are pieces of information exactly as they appear in the text, and abstractive summaries where the summary contains the semantic meaning but might use different phrasing and words than the text itself. We focus on abstractive summaries, which are more challenging because they are not necessarily made from the same words in the original text. A popular metric for evaluating a summary is ROUGE (Lin, 2004) which requires a reference summary to compare the generated summary to, and gives an evaluation based on different overlapping methods. ROUGE faces problems with correctly evaluating abstractive summaries. One problem is factuality assessment of abstractive summaries, where summaries with incorrect information may receive a relatively high ROUGE score. In (Zhang, 2020) they designed a new metric that combines ROUGE with BERTScore and FACTScore and showed that it works well for examples that already receive high ROUGE scores (easy examples) and is not as accurate on more complex examples. (Suleiman and Awajan, 2020) address the difficulties of abstractive summaries by changing the training and fine-tuning of models, and not by addressing the faults of the metric itself. (Maynez et al., 2020) researched factual and hallucination problems in model-based summaries and created annotated datasets based on human evaluation over the XSUM dataset.

ROUGE is widely used today to evaluate models in summarization tasks despite its flaws mentioned above. We aim to improve ROUGE with regards to these problems without greatly impairing the running time. An improvement in ROUGE could make a big difference in training and evaluating models for summarization tasks and therefore will be useful. Our work was based on the annotated XSUM dataset by (Maynez et al., 2020). Our methodology was to consider a regularization term to the ROUGE metric in order to improve evaluation results. Specifically we considered 'bad samples' which are samples that are non-factual yet achieved high ROUGE scores, and specifically high R1 scores. We considered 'good samples' as samples that are factual summaries which received high ROUGE scores. Our two main types of regularizations were: 1. text-overlapping between the summary and the original text. 2. semantic relations between the summary and the text or the reference summary by using dot-products on pre-trained word-embeddings. We measured our results by calculating the change of bad samples and good samples between their original ROUGE score and our metric's score. We considered a result as positive if the negative change in bad samples was greater than the negative change in good samples, or the positive change in bad samples was smaller than the positive change in good samples. We argue that bad samples are semantically further to references than good samples, thus regularizing using semantic relations with reference summary and text, or extractive with the text, will improve the credibility of the results. In the following sections we will elaborate on good and bad samples classification and show our different regularization methods, our results and our analysis of them.

2 The Data

In this section we will describe the dataset we used and elaborate on good and bad samples.

2.1 XSUM Dataset

The XSUM dataset (Narayan et al., 2018) contains news articles and short abstractive summaries that describe the articles. The summaries in this dataset are extremely short, and aim to contain only the main ideas of the text. We used (Maynez et al., 2020) annotations in order to label samples as good or bad in terms of factuality. Their annotations contain over 5000 samples from the XSUM dataset that were labeled by humans as factual or not.

2.2 Examples

For our purposes, a good sample is a sample that received a high R1 score and was annotated as factual, and a bad sample is a sample that received a high R1 score and was annotated as not factual. We refer to a high R1 score as a score higher than 0.45, as in (OpenAI). A sample can be bad because of numerous reasons, such as number faults, bad locations, bad names, bad grammar or other misinformation. Examples for good and bad samples appear in table 1.

3 Metrics with Regularization

In this section we will present our metrics using different regularization factors used in the experiments.

3.1 Text Overlap Method

This method multiplies the ROUGE score (e.g. R1) with a regularization term. The regularization term is computed with the text itself and not with the reference summary. We considered two different types of overlapping: unigram and bigram. The regularization term should capture correct information in the model summary that does not appear in the reference, causing the ROUGE score to be lower than it should be. In factual model summaries, if there is information that does not exist in the reference summary, we expect it to appear in the text, enlarging the regularization term. In non-factual model summaries, we expect some information that is not present both in the reference summary and in the text, causing the regularization term to be smaller and the score to be significantly lower than the original ROUGE score.

Denote by T the text, R the reference summary, M the model summary and |M| the length of the model summary in words. For unigrams, w is a single word, and for bigrams it is a pair of adjacent words. Denote p a real number.

The metrics are calculated in the following way:

$$\text{score}_{R1-TO}(\text{M, R, T}) = \text{R1} \cdot \text{reg}_{text}(M,T)^p$$

$$\text{reg}_{text}(\text{M,T}) := \frac{1}{|M|}\sum_{w \in M} \mathbf{1}_{w \in T}$$

We expect good-samples' scores to drop less than bad-samples' scores.

A downside of this method is that abstractive summaries might contain words not present in the text itself (like synonyms), causing the regularization term to drop even for good samples. For those cases we expect better results using the dot-product method shown next.

3.2 Semantic Relation Methods

These methods use pre-trained word embeddings to utilize the notion of similarity using dot-product. We used word2vec (Mikolov et al., 2013) representations trained on wikipedia (Yamada et al., 2020). For this approach we considered a range of n-grams (usually between 1 and 4) and calculated the regularization score by taking the maximum

	id	ref_summary	model_summary
ref_R1			
0.850000	34032798	A nine-year-old boy is being treated in hospital after being hit by a vehicle in North Lanarkshire on Saturday.	a four-year-old boy is being treated in hospital after being hit by a car in north lanarkshire.
0.604651	36462386	A farmer told to demolish a mock Tudor castle that was built without planning permission has vowed to rebuild "the work of art" elsewhere.	a farmer who built a mock-tudor castle without planning permission has failed to demolish his mock-tudor castle.
0.600000	36880863	A famous white humpback whale has been spotted on his annual migration to Australia's north.	a humpback whale has been spotted off the coast of australia's north coast.

Table 1: Hand picked samples from the annotated XSUM dataset. The first two rows are bad samples and the last one is a good sample.

dot-product value for every n-gram in the summary with respect to the text or reference. Unlike the previous method (text overlap), this method takes into account the semantic meaning and similarity of n-grams. We tried both multiplicative and additive methods.
We will use the same notation as in section 3.1, and w*w' to be the dot product between the vectors representing the n-grams w and w'.

$$\text{score}_{R1-SR-Ml}(\text{M, R, T}) = \text{R1} \cdot (\text{reg}_{SR}(M,T))^{p}$$

$$\text{score}_{R1-SR-Ad}(\text{M, R, T}) = \text{R1}+(\text{reg}_{SR}(M,T))^{p}$$

$$\text{reg}_{SR}(\text{M,T}) := \frac{1}{|M|}\sum_{w \in M} \max_{w' \in T}(w * w')$$

We note that specific word embeddings affect this method, so different word embeddings might lead to different outcomes.

4 Experiments and Results

This section describes the experiments and results using the above methods. [1]

4.1 Evaluation Methods

We use two evaluation methods: An average evaluation and a summation of evaluation scores. For average evaluation we compute evaluation scores for every annotated sample, by subtracting the new metric score from the original ROUGE score. We take a positive sign for bad samples and a negative sign for good samples. We then average these scores separately for bad samples and good samples and return the difference between them. A positive result means that bad samples' scores diminished more significantly than good samples' scores. For summation evaluation we take the same amount of good and bad samples. In our case we have more bad samples, thus we randomly select samples from the bad samples pool. We then compute the evaluation score for each sample as stated above and return the summary between these scores. Results may vary from the randomness and since the summation evaluation scores are not scaled.

4.2 Text Overlap Results

We present the evaluation results for the text overlap method. We use $p = \frac{1}{2}$ and normalize the random evaluation results by executing it 1000 times and averaging over the results. We compute scores for R1, R2 and RL for both unigram and bigram text overlap methods. We filter samples with high ROUGE scores, in particular R1 higher than 0.45. The results on hand-picked samples are shown in Table 2. Experiment results on all annotated data are shown in Table 3.

[1] The code needed to recreate the experiments can be found in this github repository: https://github.com/ohadcode/NLP/tree/main

Classification	n-gram	Average Penalty
bad	unigram	0.142650
	bigram	0.454958
good	unigram	0.104990
	bigram	0.367865

Table 2: Average Penalty for different metrics on hand-picked samples. A result is considered as an improvement when the penalty score for the bad samples is larger than the good samples' score.

Rouge Metric	n-gram	Evaluation	Score
R1	unigram	random	0.134
		avg	0.001
	bigram	random	-2.672
		avg	-0.015
R2	unigram	random	-0.894
		avg	-0.005
	bigram	random	-3.168
		avg	-0.018
RL	unigram	random	-0.720
		avg	-0.004
	bigram	random	-4.562
		avg	-0.026

Table 3: Evaluation scores for text overlap methods on all annotated XSUM dataset. An improvement is obtained when the score is positive.

4.3 Semantic Relation Results

We present the evaluation results for the semantic relations method (that uses dot-product between word vector embeddings). There are three experiments. First we test the multiplicative regularization factor for the model summary against the text in two ways - one computes the average change between bad samples and good samples, and the other computes the overall change for a randomly selected set of bad samples and all of our good samples. These are the same evaluation methods as in the text overlap experiments. We evaluate several exponent values for the regularization factor, between 0.5 and 2. For each of them we compute the evaluation scores on unigrams to 4-grams. Results are shown in Figure 1. Results for our hand picked samples are shown in Table 4. Second, we test the multiplicative regularization factor for the model summary against the reference summary (without the text), in the same two ways as the first experiment. Results are shown in Figure 2. Results for our hand picked samples are shown in Table 5. Lastly, we test an additive regularization factor. We add the regularization factor to the ROUGE score as described in section 3.2 (R1-SR-Ad). We tested on 1-gram to 4-gram against both the text and the reference to compare between them. We used p=1 (no exponent for the regularization factor). The results are shown in Tables 6 and 7.

5 Analysis

5.1 Text Overlap Analysis

We see a clear improvement in the hand picked samples for both unigram and bigram regularizations. Note that we know they are labeled correctly, but they are only a small set of samples from the entire dataset. For the entire annotated dataset, we obtain an improvement for the R1 metric by using the unigram regularization. The unigram regularization gives a large weight for words that were extracted from the text, yet summaries with misinformation tend to use words not present in the text and thus get penalized. There are good samples that use words extracted from the text (even in abstractive summaries) and hence they will have a larger regularization factor and will get penalized less. We observe that the bigram scores are negative, meaning they don't help us differentiate between good and bad samples. In our experiments we noticed that lower exponents may give better results, hence we used $p = \frac{1}{2}$ here. The regularization factor is equal or less than 1, hence larger p results in stronger penalties because the regularization factor is significantly smaller, so we believe that p should be relatively small in order for the ROUGE score to be the significant factor in the total score.

5.2 Semantic Relation Analysis

5.2.1 Multiplicative Semantic Analysis

The dot-product factor described in section 3.2, gives more weight to semantic meaning rather than exact matching, hence we expect bad samples (which might be unrelated to the text or reference summary i.e. may contain non-factual information) to suffer a greater penalty than the good samples. We indeed see that combining the results of the original ROUGE score with the semantic-related regularization factor gives positive evaluations scores, meaning an improvement in the metric's ability to

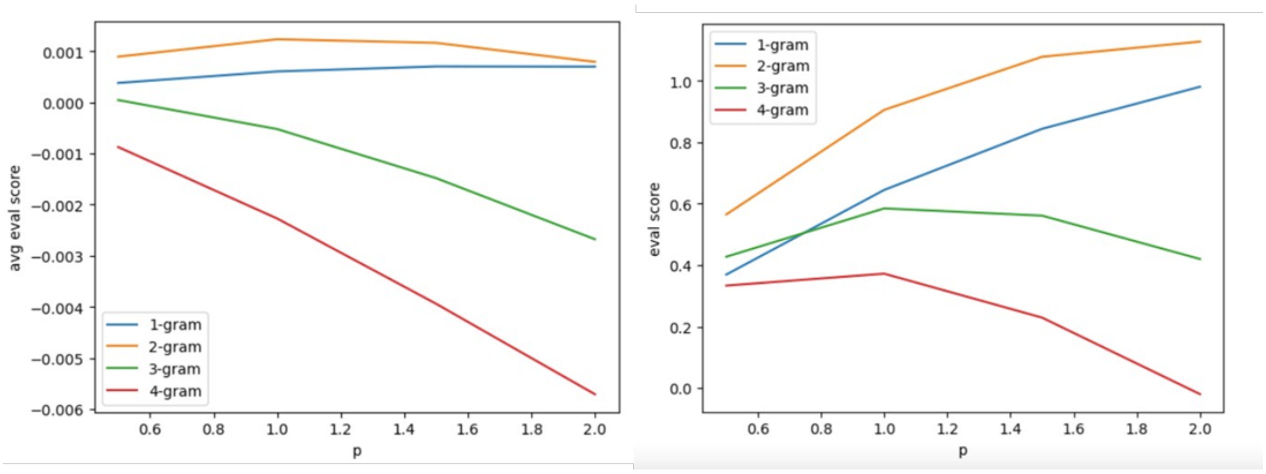

Figure 1: Evaluation scores for multiplicative regularization on text. Right plot shows total evaluation score as a function of the exponent p. Left plot shows the average evaluation. Positive evaluation scores show an improvement.

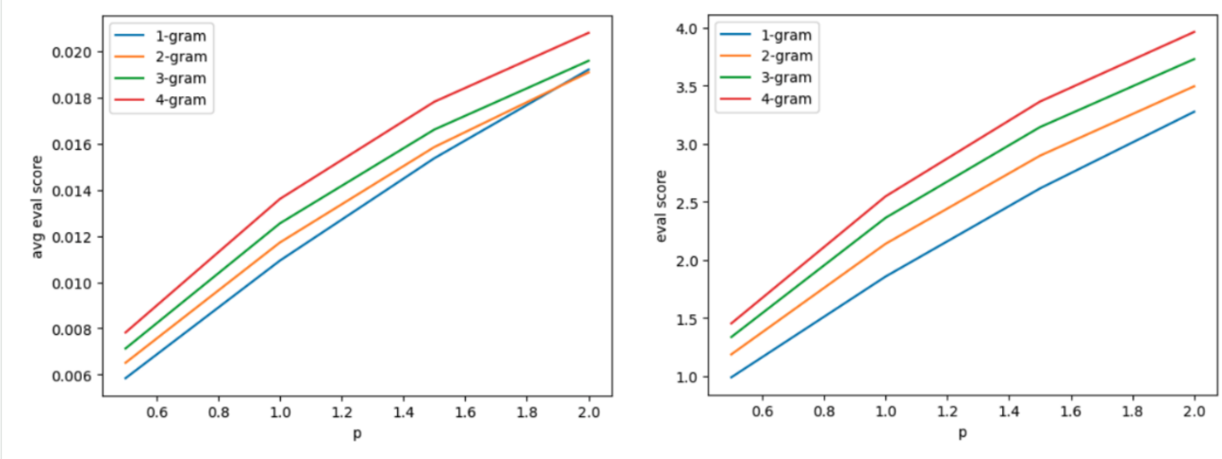

Figure 2: Evaluation scores for multiplicative regularization on reference summary. Right plot shows total evaluation score as a function of the exponent p. Left plot shows the average evaluation. Positive evaluation scores show an improvement.

differentiate between good and bad samples and hence in the overall credibility of the evaluation.

In the first experiment we use the original text as a semantic reference, and we see an improvement for unigrams and bigrams, and that the best results are around $p = 1$. We also note that while a positive evaluation score means that an improvement is achieved, the improvement is overall small. We see that if the regularization factor is too significant (when p is relatively high) we lose the advantages given by the regularization and affect the ROUGE score too much. This makes sense as we know that semantic meaning relation is important, but as p grows larger less weight is given to the ROUGE score, hence to the relation between the model summary and the reference summary. As we use bigger n-grams, we see that results are descending quickly. This could be because when we use a larger amount of words together, for a small summary, we look for a similarity between the text and the model summary. A factual abstractive summary isn't required to contain full length sentences from the text, or sentences that have a similar semantic meaning to the text.

For our experiment with semantic relation between the the model summary and the reference summery, we see that the higher the exponent p and the n-grams, the better the results. Despite that, if the regularization factor's exponent is too large, we give too much weight to the semantic meaning, until we reach a point where ROUGE and it's advantages are not significant at all in the results we receive. For larger n-grams, we capture more and more similarity in the semantic meaning between the two summaries, giving us better evaluation scores. We note that while using bigger n-grams might seem

		Penalty p=0.5	Penalty p=1	Penalty p=1.5	Penalty p=2
Classification	n-gram				
bad	1	0.049514	0.094536	0.135529	0.172906
	2	0.096627	0.179118	0.249629	0.309977
	3	0.120398	0.219300	0.300670	0.367728
	4	0.137439	0.247520	0.335831	0.406801
good	1	0.029945	0.057397	0.082612	0.105816
	2	0.064285	0.120930	0.170916	0.215094
	3	0.088286	0.162563	0.225194	0.278134
	4	0.106563	0.193373	0.264275	0.322346

Table 4: Average penalty scores of hand picked samples for the model summary against the text. The penalty scores are computed as the difference between the R1 score and our multiplicative evaluation score. An improvement for specific n and p is obtained when the penalty score for the bad samples is larger than the good samples' score.

		Penalty p=0.5	Penalty p=1	Penalty p=1.5	Penalty p=2
Classification	n-gram				
bad	1	0.019850	0.038773	0.056818	0.074034
	2	0.030565	0.058894	0.085174	0.109579
	3	0.037595	0.071733	0.102773	0.131034
	4	0.043021	0.081356	0.115585	0.146209
good	1	0.032660	0.063487	0.092590	0.120072
	2	0.048646	0.093238	0.134138	0.171672
	3	0.059882	0.113428	0.161367	0.204343
	4	0.068674	0.128838	0.181654	0.228111

Table 5: Average penalty scores of hand picked samples for the model summary against the reference summary. The penalty scores are computed as the difference between the R1 score and our multiplicative evaluation score. Results are interpreted the same as in Table 4.

n-gram	Eval Type	Text Score	Ref Score
1	regular	1.829714	3.675734
1	average	0.010730	0.021324
2	regular	2.769593	4.007365
2	average	0.016179	0.023389
3	regular	2.755364	4.336297
3	average	0.015758	0.025308
4	regular	2.501450	4.649142
4	average	0.014552	0.027249

Table 6: Evaluation for the additive regularization method against text and reference summary for unigram to 4-gram. A positive result means an improvement.

to be better, there is an important trade-off regarding running time. As we treat each n-gram as a $n * d$ vector, d is the word-embedding dimension, we a have multiplicative dependency in n in the function's running time, which might become significant in large datasets or for large texts. Hence, the results of larger n-grams in our experiments should be treated carefully. We notice that while we achieved improvements for both model vs text and model vs reference for the evaluation on the entire annotated dataset, we got negative results for the model vs ref in our hand picked samples. Assuming that the entire annotated dataset is more reliable, we still consider our results to be positive. We note that the experiments are shown for regularization on ROUGE R1 scores, but the general ideas and methods described above are true for R2 and RL as well, although results may vary.

5.2.2 Additive Semantic Analysis

We obtain positive results for both evaluations on model summary against the text and model summary against the reference in the additive regularization, meaning that like the multiplicative options, this method also shows an improvement. Specifically, we notice that the score for the model summary against the reference summary achieved bet-

Class	n-gram	Eval Text	Eval Ref
bad	1	0.866382	0.937897
	2	0.747760	0.906527
	3	0.690952	0.885912
	4	0.652256	0.870062
good	1	0.912017	0.896758
	2	0.816025	0.847624
	3	0.752103	0.813595
	4	0.702977	0.787574

Table 7: Evaluation for the additive method on our hand picked samples. We interpret an n-gram as an improvement if it's good score is higher than it's bad score.

ter results than against the text, as shown in Table 6.

We note that while the running time of this method will be similar to the running time of the multiplicative method, as the dominant factor is the dot product calculation, the evaluation scales will not be the same as in the results shown in Figures 1 and 2. While the maximal possible ROUGE score is 1, and so is the maximal possible score of the multiplicative method, for the additive method results my vary up to a perfect score of 2. Normalizing is a possibility.

6 Conclusions

The ROUGE metric is widely used for evaluating model generated summaries against a reference summary. As described earlier, it faces difficulties with evaluating abstractive summaries. Our goal was to improve upon ROUGE in evaluating abstractive summaries, while keeping its strengths. We claim that the ROUGE metric can be improved by adding a regularization factor based on the text or the reference, which is computed by text-overlap or by semantic relation methods. Some of the methods require more running time than others, and improve the credibility of the metric in different ways. We see that semantic methods tend to improve the metric more than the text-overlap method. In particular, the semantic method against the reference seemed to show the best results. As ROUGE is widely used, this improvement can be significant in the future in evaluating summaries, and in improving the quality of summaries created by models. For future work, the metrics stated above should be tested on other annotated datasets, especially ones that test factuality. In addition, an accurate estimation for differences in running times between our methods and the original ROUGE should be performed. Finally, these methods could be tested on good model generated summaries that received low R1 scores.

References

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization.

Tomas Mikolov, Kai Chen, Kai Chen, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization.

OpenAI. Rouge metrics for summary headline.

Dima Suleiman and Arafat Awajan. 2020. Deep learning based abstractive text summarization: Approaches, datasets, evaluation measures, and challenges.

Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2020. Wikipedia2vec: An efficient toolkit for learning and visualizing the embeddings of words and entities from wikipedia.

Yuhui Zhang. 2020. A close examination of factual correctness evaluation in abstractive summarization.