國立臺灣師範大學理學院
資訊工程學系
碩士論文

Department of Computer Science and Information Engineering
College of Science
National Taiwan Normal University
Master's Thesis

Diversity and Quality: Comparing Decoding Methods with PEGASUS for Text Summarization

唐科南
Keenan Nathaniel Thompson

指導教授:陳柏琳 博士
Advisor: Berlin Chen, Ph.D.

中華民國110年10月
October 2021

Abstract

This thesis offers three major contributions: (1) It considers a number of diverse decoding methods to address degenerate repetition in model output text and investigates what can be done to mitigate the loss in summary quality associated with the use of such methods. (2) It provides evidence that measure of textual lexical diversity (MTLD) is as viable a tool as perplexity is for comparing text diversity in this context. (3) It presents a detailed analysis of the strengths and shortcomings of ROUGE, particularly in regard to abstractive summarization. To explore these issues the work analyzes the results of experiments run on the CNN/DailyMail dataset with the PEGASUS model.

Keywords: summarization, diverse decoding, PEGASUS, ROUGE, lexical diversity

Table of Contents

1. Introduction

1.1 Background and Motivation

While it is increasingly the trend in natural language processing (NLP) research, particularly when considering work from major universities, technology companies, and other research institutions with access to near unlimited amounts of computing power, to focus on training models for longer on larger and larger collections of text, this is simply not a realistic option for most individuals. Though services like Google Colaboratory (Colab) have made computing power in the form of GPUs (and other types of processors that similarly improve training speeds like TPUs) more accessible to the average person, users of these platforms (even paid users) still face serious limitations both in terms of the number of processors they have access to and amount of time they can use these processors. This fact means that there is a world of difference between the considerations of an individual NLP researcher and a company with the millions of dollars required to run hundreds or thousands of GPUs for weeks to train a model on some colossal dataset.

Luckily, thanks to the open nature of research in the field, individuals can benefit greatly from publicly available checkpoints to warm-start models. Afterward, the individual is free to further fine-tune the model to suit any relevant, downstream task. This offers tremendous savings in terms of both time and computing power, and it makes it so that individuals, even those with access to the most meager of resources, can easily reproduce state-of-the-art results across a wide array of NLP tasks and run experiments with world-class models.

Though automatic text summarization has made great strides, particularly in recent years with the advent of transformer-based models, current methods still suffer from several, notable

deficiencies. They still struggle to deal with long documents or multiple documents, and the vocabulary of machine-generated summaries tends to be significantly less diverse than human-generated summaries. Furthermore, even some state-of-the-art models can produce summaries that degenerate into a single, repeated phrase, and sometimes the generated summary is inaccurate, either misrepresenting information from the source text or wholly inventing new information (Holtzman et al. 1).

This thesis focuses on one of the aforementioned issues, repetitive output text. The contributions of this work are three-fold: (1) It considers a number of diverse decoding methods to address degenerate repetition in model output text and investigates what can be done to mitigate the loss in summary quality associated with the use of such methods. (2) It provides evidence that measure of textual lexical diversity (MTLD) is as viable a tool as perplexity is for comparing text diversity in this context. (3) It presents a detailed analysis of the strengths and shortcomings of ROUGE, particularly in regard to abstractive summarization.

2. Related Works

2.1 Approaches and Models

2.1.1 Extractive, Abstractive, and Hybrid Approaches

Approaches to automatic text summarization can be divided into two major categories, extractive and abstractive. Extractive approaches attempt to first identify salient sentences from the source text and later assemble them into a summary, while abstractive approaches attempt to create a summary by rephrasing information from the source text, generating novel passages in the process. Extractive summarization is the easier of the two approaches, as copying sentences from the source text guarantees both a certain, basic degree of accuracy and that the output text will be grammatically correct, but abstractive summarization is, for several reasons, often seen as more desirable. First, abstractive approaches mirror the way humans actually summarize documents, and additionally, more sophisticated summarization techniques like paraphrasing are only possible with an abstractive approach. Despite this, because of the degree of difficulty associated with abstractive approaches, extractive approaches have dominated summary research throughout most of the history of the field. But, in recent years, a number of innovations have made abstractive approaches increasingly viable (See et al. 1–2).

In the paper *What Have We Achieved on Text Summarization* researchers used ROUGE and PolyTope, a manual summary evaluation framework they developed themselves based on Multidimensional Quality Metric (MQM), on a number of different models to offer a fine-grained analysis of what extractive and abstractive approaches get right and what they get wrong (Huang et al. 4). While they found no significant gap between extractive and abstractive methods with regard to ROUGE, when evaluating with PolyTope, at the most basic level, they

found that “extractive summarizers are in general better than their abstractive counterparts thanks to strength in faithfulness and factual-consistency (Huang et al. 1).”

They determined the main flaws of each approach to be “unnecessity for extractive models, and omission and intrinsic hallucination for abstractive models (Huang et al. 2).” In this context “unnecessity” means that the generated summary includes sentences from the source text that are not particularly relevant, and “intrinsic hallucination” means that terms or concepts from the source text are misrepresented in the summary (Huang et al. 5). Altogether they found that extractive approaches suffer from significantly fewer errors. Extractive approaches tend to only make 3 kinds of errors—addition (extraneous information), omission, and duplication—whereas abstractive approaches commonly experience 4 to 7 types of errors (Huang et al. 6). Extractive methods are significantly better in terms of accuracy — since they directly copy from the source text, this is as expected — but they are no better than abstractive approaches with regards to addition and duplication. Both approaches are comparable in terms of fluency, as recent neural, abstractive models are capable of forming cohesive summaries in a way that abstractive approaches were previously unable to match (Huang et al. 6).

In analyzing the source sentences used for generation by abstractive models, they found a tendency for them to ignore sentences toward the middle or end of the document. This suggests that the performance of abstractive approaches may be heavily influenced by the leading bias (the tendency to include important information near the beginning of the document, common in news articles) of the source text (Huang et al. 8).

Hybrid models, they found, tended to reflect the respective strengths and weaknesses of each approach. Their study only included one hybrid model (BottomUp). Though it produced strong ROUGE scores, it ranked the second worst on PolyTope, and it tended to experience more

problems with accuracy than other models. The strength of this approach is that this hybrid process, extracting then rewriting sentences from the source text, leads to better recall. Its weakness is that the abstractive generative model limits its attention to the sentences in the intermediate, extractive summary, and if this extractive summary omits some important information, this can result in an inaccurate or incomplete final, abstractive summary (Huang et al. 7).

2.1.2 PEGASUS

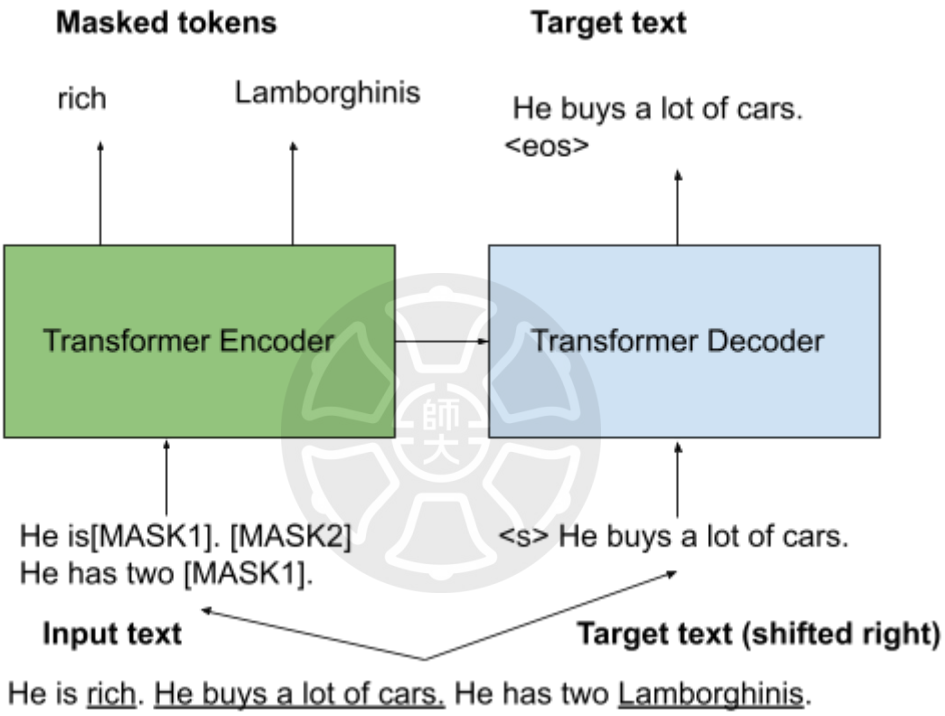
PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization) is a model first proposed by Google Brain in a 2020 paper. In terms of its architecture, PEGASUS is a standard Transformer encoder decoder, and as with BERT (an earlier model from Google), its advancements come from its particular pre-training goals.

BERT's masked language modeling (MLM), inspired by the Cloze task, involves masking a certain percentage of tokens at random and then later predicting them. BERT specifically masks (replaces with a [MASK] token) 15% of input tokens. However, in order to avoid creating a mismatch between pre-training and fine-tuning (because the [MASK] token will not appear in fine-tuning), the tokens are not always masked. Of the 15% of tokens randomly selected from the input, 80% are replaced with the [MASK] token, while 10% are replaced with a random token, and the remaining 10% are unchanged.

PEGASUS combined BERT's strategy of masking tokens with masking sentences. It pre-trains on these tasks on either the 750 GB C4 dataset of text from 350 million websites or HugeNews, a dataset introduced here, composed of text from 1.5 billion news and news-like websites (Zhang et al. 4). The researchers hypothesized that the sentence masking and generation objective would be especially suitable for abstractive summarization because of how much it

resembles the downstream task (Zhang et al. 2). This process, which they refer to as gap sentences generation (GSG), leads to incredible performance in downstream tasks and it works particularly well when important sentences are chosen to be masked, as opposed to leading or random sentences (Zhang et al. 2).

2.1.2.1 Diagram | PEGASUS Architecture



While this diagram shows pre-training with both MLM and GSG, as explained below MLM was abandoned in training PEGASUS Large (Zhang et al. 1).

PEGASUS achieves human-level performance on several datasets. Moreover, it boasts such performance without needing too much in the way of fine-tuning. After fine-tuning on a paltry 1000 samples from 6 datasets, PEGASUS surpassed the state-of-the-art on all of them. The gap sentences ratio (GSR - ratio of gap sentences to other sentences in the document) or

mask rate proved to be of great importance. If the ratio is too low then the task is not sufficiently challenging, but conversely if it is too high, the model will not have enough context with which to inform its generations. The optimal value for GSR varied significantly based on the dataset, but in all cases the best performing ratios were all below 50%. On CNN/DailyMail the model with a 15% GSR provided the best result, and for XSum/Reddit TIFU and Wikihow, the best performing models had GSRs of 30% and 45% respectively (Zhang et al. 5–6).

Though masked language MLM was used (in addition to GSG) in pre-training PEGASUS Base (223 million parameters), when it was found to inhibit gains with more pre-training steps (500 thousand), it was abandoned for PEGASUS Large (568 million parameters) (Zhang et al. 6). PEGASUS Large has an effective GSR of 30%. As is the case with MLM, GSG for PEGASUS Large does not always mask selected sentences. In order to encourage the model to copy, which is important for some datasets, 20% of the selected sentences are left unchanged, and the GSR is increased to 45% to have a similar number of gap sentences as the with the 30% ratio deemed optimal by the researchers through their experiments (Zhang et al. 6).

2.2 Metrics

2.2.1 Summary Quality

In the field of machine summarization, ROUGE is the primary metric by which researchers evaluate summary quality. Kavita Ganesan's 2018 paper *ROUGE 2.0* offers a good, brief explanation of the metric. "ROUGE, or Recall-Oriented Understudy for Gisting Evaluation is a method to automatically determine the quality of a summary by comparing it to another set of (ideal) summaries often created by humans (Ganesan 1)." More specifically ROUGE is

calculated by counting the number of words or n-grams that the ideal and generated summary share in common (Ganesan 1).

Within ROUGE, there are several kinds of scores. Today models generally provide two ROUGE-N scores and a ROUGE-L score as points of comparison to other approaches in the field. ROUGE-N measures n-gram co-occurrence, where n refers to the length of the n-gram. So, ROUGE-1 measures unigram overlap, and ROUGE-2 measures bigram overlap. Consequently, these are the two ROUGE-N measures of interest in summarization research. ROUGE-L measures the longest common subsequence (LCS) between the ideal and generated summary.

2.2.1.1 Formula | ROUGE-N

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad \text{(Lin 1)}$$

2.2.1.2 Formula | ROUGE-L

$$R_{lcs} = \frac{LCS(X,Y)}{m}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n}$$

$$F_{lcs} = \frac{(1+\beta^2) R_{lcs} P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}}$$

Where there are two summaries, X and Y or length m and n respectively, and $\beta = P_{lcs}/R_{lcs}$ (Lin 2).

The 2004 paper that introduced ROUGE, *ROUGE: A Package for Automatic Evaluation of Summaries,* demonstrated solid correlation between ROUGE and human evaluations, but different ROUGE variants showed higher correlation depending on the task. For summarization of 100 word-long, single documents, of the ROUGE-N group, ROUGE-2 proved to have the best correlation, and ROUGE-L also performed well. For very short, single documents, ROUGE-1 and ROUGE-L performed well, but ROUGE-2 performed poorly (Lin 5).

Ideally a referenced-based metric for summary quality should explain how much information is shared between two summaries. The 2020 paper *Understanding the Extent to Which Summarization Evaluation Metrics Measure the Information Quality of Summaries* examines if this is actually the case with metrics like ROUGE, or if they instead measure some other, less-desirable, latent quality, like whether two texts simply discuss the same topic (Deutsch and Roth 2). To do this the researchers measured the correlation between ROUGE and yet another metric, Pyramid Score. This approach assumes Pyramid Score as the gold standard, but because the methodology behind Pyramid Score relies on exhaustive annotation of summary content units (SCUs - a phrase that shares some particular bit of information) by domain experts and is completely based on how much information a reference and evaluated summary share (measured by SCU overlap), this is a reasonable assumption.

For ROUGE-1 they found that, on average, only 25% of its score could be derived from SCU overlap with a reference summary. Later in the study, considering tokens by category, they found that most of ROUGE-1 could be explained by the degree to which the reference and evaluated summary discussed the same topic (Deutsch and Roth 6). When they compared the correlation of several other summarization metrics to ROUGE-1 and to Pyramid Score, they found that many shared a much higher correlation with ROUGE-1, and at best their correlation

with PyramidScore was around the same of that of ROUGE-1 with PyramidScore, approximately 0.6(Deutsch and Roth 7). This suggests that these metrics may suffer from the same issues as ROUGE-1, and are similarly poor evaluations of information overlap.

In addition to its shortcomings in terms of measuring information overlap, ROUGE's reliance on n-gram overlap causes other issues. Its dependence on n-gram overlap makes ROUGE a largely unsuitable metric for abstractive summaries which involve significant paraphrasing. It makes no consideration for synonymous words or concepts (Ganesan 1). Further, ROUGE may unfairly penalize shorter summaries. If the generated summary is shorter than the reference summary, it has less potential for n-gram overlap. This means its score will be impacted, regardless of how well it captures the ideas from the reference summary. It could be the case that the generated summary captures the same ideas as the reference summary while cutting back on some verbosity, but ROUGE does not allow for such fine-grained analysis (Ganesan 1). Finally, ROUGE does nothing to measure grammaticality, accuracy, or fluency (Ng and Abrecht 1).

Other research has produced results that are at least somewhat supportive of ROUGE. Though researchers from Zhejiang University found poor correlation between ROUGE and human judgement (ROUGE-1: 0.40 ROUGE-2: 0.32, ROUGE-L: 0.32) at the instance-level (single sample), at a system-level (aggregation of samples), they found relatively high correlation (ROUGE-1: 0.78 ROUGE-2: 0.73, ROUGE-L: 0.52). But, even at the instance-level they found that ROUGE could be used to guarantee a certain degree of accuracy and fluency (as measured by their PolyTope framework), and of the three scores, ROUGE-2 was best at evaluating fluency (Huang et al. 9).

2.2.2 Lexical Diversity

Lexical diversity (LD) refers to the number of different words used in a given text. In this context a given text composed of a wider array of different words than another text is deemed to be more diverse (McCarthy and Jarvis 381). LD indices have long been used in a number of fields to measure everything from early-stage Alzheimer's disease to one's socioeconomic status.

Though there is no doubt of their usefulness, the problem that many of these indices face is their sensitivity to text length (McCarthy and Jarvis 381). This is because of how many indices deal with two key metrics, tokens and types, and the ratio between them (token type ratio - TTR). The tokens are the number of words in the text, and the types are the number of different words. As token count increases, type count steadily declines. With each additional token, there is a decreasing likelihood of encountering a new type. This is explained by a property of language, that is to say that any text of a significant length cannot be meaningful without repetition of tokens (McCarthy and Jarvis 382). But this increase in token repetition does not necessarily mean any loss of diversity in what a reader might perceive. An inability to properly account for this fact led to many researchers to report diversity scores that were "confounded with text length (McCarthy and Jarvis 382)."

One benefit to this sensitivity to text length, however, is that the aforementioned gradual decrease in new types can be used to indicate a text's thematic saturation, the point at which new types are no longer encountered and all the types that are representative of the text's theme are present (McCarthy and Jarvis 382). This is a useful metric because it allows researchers to understand that a text is sufficiently long enough to apply their diversity calculation function. The calculation of the MTLD (measure of textual lexical diversity) index makes use of a concept close to this, in addition to insights from another index, mean segmental TTR (MSTTR), as the

rationale for its calculation. MTLD calculates the mean length of sequential word strings that maintains a certain TTR (0.720). The TTR of each word in the text is calculated sequentially, and when the default TTR factor size (0.720) is reached, the factor count increases by 1, and the TTR calculations reset. Finally, the total number of words is divided by the total factor count. MTLD is processed twice, once forward and once backwards, and the mean of these values gives the final MTLD index.

LD indices are graded in terms of four kinds of validity: convergent validity, divergent validity, internal validity, and incremental validity. Convergent validity relates to how much results from a given index agree with other accepted, standard indices. Conversely, divergent validity measures how much results from an index disagree with indices seen to be flawed. Internal validity measures an index's sensitivity to text length, and incremental validity refers to how much more information an index gives, relative to other, similar indices(McCarthy and Jarvis 388–89). MTLD performs well with regards to all four, and it is the only index that does not vary as a function of text length (McCarthy and Jarvis 381).

2.3 Neural Text Degeneration

Though maximum likelihood training has proven to be a useful tool to train language models applicable to many NLP tasks, decoding model outputs based on maximum likelihood methods has delivered less-than-ideal outcomes (Holtzman et al. 1). Output text generated by such maximization techniques, like beam search, often result in what researchers at the University of Washington refer to as degeneration, "output text that is bland, incoherent, or gets stuck in repetitive loops (Holtzman et al. 1)." This is caused by a strange phenomenon. Despite the fact that state-of-the-art models do, as one might expect, assign higher probability to more

coherent, human-like sequences, the highest scores for longer sequences are often boring and repetitive (Holtzman et al. 2).

As with approaches to summarization, decoding algorithms can be separated into two camps: deterministic and stochastic. A deterministic algorithm selects each token at a given time step based on some established rule. The most naive approach among this group, greedy decoding, simply selects the most probable token at a given step (Welleck, Kulikov, Kim, et al. 3). Beam search is another deterministic algorithm, and it has become ubiquitous as the go-to decoding algorithm for decoding model outputs in NLP. It works by performing a breadth-first search to approximate finding the most likely sequence over some restricted search space (Ippolito et al. 3). At each decoding time step, it considers *b* candidates, and it then explores every path from the set of *b* to find the next candidates, eventually choosing the beam with the highest score. Log-likelihood is typically used to score each partial sequence. Because beam search only explores a narrow band of the space of possible sequences, it is not conducive to delivering diverse outputs, and it usually generates only slightly different variations of the same high probability sequences (Ippolito et al. 3).

Stochastic decoding algorithms depend on a degree of randomness. Pure samping, the most naive stochastic approach, involves sampling directly from the probabilities predicted by the model, and it generally results in incoherent text. The team from the University of Washington blames this on an “unreliable tail” of tens of thousands of candidate tokens that are, despite their individually low probabilities, over-represented on the whole (Holtzman et al. 2). A number of approaches have been suggested to moderate the high variance of decoding with stochastic sampling in order to produce reasonable outputs. Top-*k* sampling is one popular such technique, and it works by limiting consideration at each decoding step to a fixed *k* number of

most probable tokens (Welleck, Kulikov, Kim, et al. 3). If the value of *k* is too low, it risks generating bland text, and if it is too large, low probability candidates will have their chance of being selected increased during renormalization. To deal with some of the potential issues with top-*k* sampling, it is often combined with temperature sampling. In this method of sampling, lowering the temperature reshapes the probability distribution towards high probability tokens, and this is done before choosing the top *k* tokens. This has proven effective in improving the quality of generated text, but it comes at the cost of diversity (Holtzman et al. 6).

2.3.1 Formula | Top-*k*

Given a distribution $P(x|x_{1:i-1})$, the top-*k* vocabulary is the set of size *k* that maximizes $\sum_{x \in V^{(k)}} P(x|x_{1:i-1})$ (Holtzman et al. 5).

2.3.2 Formula | Temperature

Given logits $u_{1:|V|}$ and temperature *t*, the softmax is re-estimated as

$$p(x = V_l|x_{1:i-1}) = \frac{\exp(u_l/t)}{\sum_{l'} \exp(u'_l/t)}$$

(Holtzman et al. 6).

Given the flaws of top *k* and temperature, the researchers propose their own stochastic decoding solution, nucleus or top *p* sampling. Nucleus sampling works by selecting the highest probability tokens above some threshold *p*. Unlike top *k* which deals with a fixed value, the value of *p* and thus the size of the set of tokens from which the method samples, can vary greatly at each step. The value of *p* is calculated dynamically as a function of "changes in the model's confidence region over the vocabulary (Holtzman et al. 6)." Using similarity to the perplexity of the human-generated, gold text as a metric, the researchers found that nucleus sampling achieved

the closest to human-level perplexity. While these decoding-time-based solutions do ultimately result in better, more diverse output text, some research has explored the possibility that the bigger issue lies in how the models are trained. Natural language does not maximize probability, so models training via a maximization task may be a mismatch for downstream tasks(Holtzman et al. 7). Some research from Facebook has shown the potential of unlikelihood training, forcing the model to assign lower probability to less likely tokens and sequences, as opposed to selecting for the most probable ones (Welleck, Kulikov, Roller, et al. 1).

2.3.3 Formula | Nucleus Sampling

Given a distribution $P(x|x_{1:i-1})$, the top-p vocabulary is the smallest set such that

$$\sum_{x \in V^{(p)}} P(x|x_{1:i-1}) \geq p$$

(Holtzman et al. 4).

Though they solve some problems, diverse decoding strategies may have drawbacks. *Comparison of Diverse Decoding Methods from Conditional Models, a* 2020 paper from the University of Pennsylvania, found a "marked trade-off between diversity and quality (Ippolito et al. 7)." More specifically it found that fluency and adequacy shared a strong negative correlation with diversity, further suggesting that such diverse methods should be avoided in tasks like machine translation where mistakes can deeply impact coherence (Ippolito et al. 7). In addition to potential problems with coherence, their findings regarding the relationship between text diversity and what readers found interesting were less-than-ideal. Though researchers expected to find a positive correlation between diversity and interestingness, they did not and concluded that "existing diversity statistics are insufficient for capturing what it means to humans for outcomes to be interesting (Ippolito et al. 7)."

3. Methodology

3.1 Environment

All experiments were conducted on Google Colab Pro with Python 3.7 and Pytorch version 1.9.0 with Cuda version 11.1 using a Tesla P100 PCIE 16GB Nvidia GPU. Models were loaded using the Huggingface transformers library version 4.11.3.

3.2 Metrics

3.2.1 ROUGE

Despite the many problems with ROUGE outlined earlier in this work, it remains a standard in NLP research, and having ROUGE scores allows for easier comparison of the approach outlined here with others in the field. Accordingly, ROUGE-1, ROUGE-2, and ROUGE-L scores are presented for all generated summaries. Hopefully, the exploration of ROUGE in the Related Works section provides useful, additional context with which to interpret these scores.

ROUGE is the python package rouge[1]. The package offers the caveat that its implementation is independent of the official ROUGE script, and so results from this script may be slightly different. All ROUGE scores presented here should be considered with this context.

3.2.2 MTLD

Though other research on diverse decoding methods has used perplexity as their metric, the experiments here use a lexical diversity index, MTLD, as a substitute. While MTLD does not

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

deal with the probability of output tokens, it can serve as an accurate metric to measure repetition in text, and this is the core quality of concern for the experiments here. Furthermore, because a method with which to calculate MTLD is readily available via a Python package, it is significantly easier to set up to measure in experiments when compared to perplexity where no such package is available (Holtzman et al. 7).

MTLD is calculated with the lexical-diversity library (version 0.1.1) available via the Python Package Index (PyPI)[2].

3.3 Dataset

Just as ROUGE is a standard metric in assessing summary quality, the CNN/DailyMail is a standard dataset for the task. It consists of articles on a number of topics from the two outlets for which it is named. Each article is accompanied by a bullet point style list of highlights from the article that is used as the ideal summary. The dataset is split into three sub-groups, training, validation, and test, and the experiments here use 100 samples from the test group to generate summaries.

3.4 Models

3.4.1 PEGASUS

The experiments here use two variations of the model, all available on Hugging Face — PEGASUS Large AND PEGASUS Large fine-tuned on CNN/DailyMail. The max length for model outputs was set to 128, keeping with the limits set for CNN/DailyMail in the original PEGASUS paper.

[2] https://pypi.org/project/lexical-diversity/

3.5 Experiments

The code to run all of the experiments can be found here[3].

3.5.1 Baselines

Because of beam search's role as the standard decoding strategy in NLP tasks, it can be treated as the baseline against which to measure the diverse decoding methods. The experiments use both the standard PEGASUS beam size of 8 and also beam size 16 to have a common point of comparison with *The Curious Case of Neural Text Degeneration*, the paper that informs the experiments in this thesis.

3.5.2 Diverse Decoding Strategies

The experiments here consider the following three diverse decoding methods: top-*k*, top-*k* with temperature, and nucleus sampling.. These methods were chosen due to their relative popularity, and these values were informed by experiments in the paper *The Curious Case of Neural Text Degeneration* (Holtzman et al. 4), though this thesis also tests with some additional values that the paper did not use. Though the aforementioned paper also included pure sampling and stochastic beam search, it was decided to exclude them here. It is well understood that pure sampling is a poor decoding method, and it seems as if the researchers behind the paper included it only for illustrative purposes. As for stochastic beam search, in my review of the literature on decoding strategies, it did not appear nearly as much as the other methods chosen for these experiments.

[3] https://colab.research.google.com/drive/11hctIIRXKKBuN3b13a32_DUQKTilNOu7

In *The Curious Case of Neural Text Degeneration,* the decoding methods deemed to be the greatest success were those that produced results for perplexity closest to human-generated text (Welleck, Kulikov, Roller, et al. 6). Here, since perplexity is replaced by MTLD, a method's level of success will be judged by its proximity to the MTLD of human-generated summaries.

4. Results and Discussion

The MTLD values closest to the human-generated summaries are bolded.

4.1 Table | Large | All

	ROUGE-1	ROUGE-2	ROUGE-L	MTLD
Actual Summaries				215.89
Beam, b = 8	36.72	10.07	33.78	114.37
Beam, b = 16	36.78	10.14	33.65	116.91
Top-k, k = 40	36.57	9.95	33.58	106.54
Top-k, k = 640	35.65	9.02	32.80	127.75
Top-k, k = 40, t=0.7	37.30	10.27	34.04	102.59
Top-k, k = 640, t=0.7	36.48	9.41	33.54	107.83
Nucleus, p = 0.95	34.64	9.18	31.99	123.62

Here one sees how drastic an effect temperature has on diversity. Looking at the two results from using top-k sampling alone (without temperature) when the value of k increases from 40 to 60, there is an over-20-point gain in terms of MTLD. Later in the table when these two values for k are repeated, but this time with t=0.7, there is only a 5-point increase in MTLD. For the non-fine-tuned PEGASUS Large, none of the diverse decoding methods came even close to the lexical diversity as the actual summaries, but for the fine-tuned model, some did come very close.

4.2 Table | Fine-tuned | All

	ROUGE-1	ROUGE-2	ROUGE-L	MTLD
Actual Summaries				215.89
Beam, $b = 8$	39.61	13.27	37.17	178.92
Beam, $b = 16$	40.37	13.58	37.91	178.78
Top-k, $k = 40$	39.28	10.97	36.99	**206.63**
Top-k, $k = 640$	39.56	11.14	37.53	232.24
Top-k, $k = 40$, t=0.7	41.39	12.81	38.61	**204.69**
Top-k, $k = 640$, t=0.7	40.85	12.91	38.97	193.19
Nucleus, $p = 0.95$	35.75	9.63	33.71	318.82

When comparing the Large and Fine-tuned results, the first thing that becomes apparent is how much fine-tuning does to improve lexical diversity. It, of course, also leads to great improvements in ROUGE scores, but this is expected. Though the exploration of this issue is largely outside the scope of this work, this leads one to wonder just how close further fine-tuning could take the model towards human-levels of lexical diversity. It is also certainly possible, as with ROUGE, lexical diversity could experience substantial diminishing returns from more training. For PEGASUS, ROUGE scores improve only slightly when the model is trained on 10,000 samples instead of 1,000(Zhang et al. 16); lexical diversity could suffer the same fate.

4.3 Table | PEGASUS Paper Results

	ROUGE-1	ROUGE-2	ROUGE-L
PEGASUS | 0 Examples	32.90	13.28	29.38
PEGASUS | 1000 Examples	41.72	19.35	38.31

(Zhang et al. 16)

Though the ROUGE-2 scores shown here are both generally low and significantly different from those in the original PEGASUS paper, the other values are fairly close. With regards to the low ROUGE-2 scores, two things should be taken into consideration here: (1) As mentioned in the Methodology section, ROUGE is calculated here using an independent implementation that may produce slightly different results (2) The team behind PEGASUS mentioned some low ROUGE scores in their paper, saying "even low-ROUGE model summaries often were high quality." The paper's appendix specifically provides an example of one such summary that is of decent quality but has a low ROUGE-2 score. In a section that begins by commenting on the shortcomings of ROUGE, particularly with regards to abstractive methods, the team behind PEGASUS added that "perplexity-optimized models using aggregated ROUGE" returned better quality summaries than models directly optimizing ROUGE (Zhang et al. 8).

4.4 Sample | Fine-tuned | $k = 40$

Actual Summaries	$k = 40$
A lawyer for Dr. Anthony Moschetto says the charges against him are baseless . Moschetto, 54, was arrested for selling drugs and weapons, prosecutors say . Authorities allege Moschetto hired accomplices to burn down the practice of former associate . "No challenge poses more of a public threat than climate change," the President says . He credits the Clean Air Act with making Americans "a lot" healthier .	Dr. Anthony Moschetto's attorney says "no evidence" supports the allegations.<n>Authorities say his client was involved in a failed conspiracy to have another doctor hurt or killed.<n>Messerto,54, is charged with conspiracy, burglary, arson and prescription sale and possession of weapons. Gloria Borger: Obama wants average Americans to focus on public health in climate change conversation.<n>president credits CleanAir Act with making Americans "a lot" healthier.<n>He did not appear particularly concerned about the current Supreme Court challenge to the Affordable Care Act.

Here is an example of output that provides a fairly decent summary, despite low ROUGE scores. This appears to support the paper's conclusions regarding ROUGE not being the be-all and end-all when it comes to judging abstractive summary quality.

4.5 Table | Fine-tuned | Nucleus Sampling

	ROUGE-1	ROUGE-2	ROUGE-L	MTLD
Actual Summaries				215.89
Nucleus, $p = 0.7$	40.61	11.81	38.62	168.27
Nucleus, $p = 0.8$	40.21	12.27	38.48	187.99
Nucleus, $p = 0.85$	38.14	10.92	36.33	**203.72**
Nucleus, $p = 0.9$	39.00	10.68	36.93	**225.63**

It was originally the intention to only experiment with nucleus sampling using the value from the original paper, but using the value of *p* provided there (0.95), resulted in surprisingly poor results. It should be noted that the experimental setup here is substantially different from that of *The Curious Case of Neural Text Degeneration.* In that paper the output text is generated with a significantly different model (GPT2) conditionally based on the initial paragraph (limit 1-40 tokens) of documents in the WebText dataset, and output is restricted to a maximum length of 200 tokens, as opposed to the 128 maximum output length used in the experiments here(Holtzman et al. 6).

Though the ROUGE scores are low as well, these scores are considerably less shocking than the MTLD values. Especially in the case of nucleus sampling with the fine-tuned model, the result is indeed far too diverse. In *The Curious Case of Neural Text Degeneration* the method with the highest diversity was pure sampling, but as mentioned in that paper, the output from pure sampling is generally very poor or even incoherent. And as discussed in the section on lexical diversity, it is a feature of language that a text of any serious length needs some repetition to be comprehensible.

The value for lexical diversity was indeed so high that it seemed it might be the result of some error. But, after re-running this experiment, the value was 313.67, which is lower but not by much. Because of these strange results, it was decided to engage in a more thorough investigation of nucleus sampling by experimenting with several values of *p* to attempt to produce results for nucleus sampling with reasonable quality and diversity scores.

After seeing the MTLD values that were much too high, it was decided to first try again with values for *p* that were significantly lower. While these values led to much improved ROUGE scores, they resulted in MTLD values that hovered around what beam score produced. But, values slightly lower than 0.95 produced results with decent ROUGE scores and levels of diversity much closer to those found in the human-generated summaries. From the table above it is clear that, with nucleus sampling, an additional 0.05 can lead to fairly substantial changes in terms of MTLD (usually adding around 20 points), but it is still unclear as to what causes the massive leap in MTLD (around a 90 point increase) when going from 0.90 to 0.95. For posterity, testing was also performed using the Large, non-fine-tuned model with these values for *p*, but because changing these values did not result in significant or interesting changes, the scores are not presented here.

Though there are some exceptions, the MTLD results are largely what one would expect from the literature on these decoding methods. Here as in *The Curious Case of Neural Text Degeneration*, the two methods (of those shared in common) with the highest levels of diversity (in terms of perplexity in that paper and in terms of MTLD here). MTLD increases with higher values of *k* and decreases with higher values of *t*, and this is the same as what one would expect with perplexity. Similarly, beam search expectedly produced relatively low MTLD scores, and nucleus sampling produced higher scores with higher values of *p,* as it should(Holtzman et al. 7).

These many points of consistency seem to suggest that MTLD approximates perplexity quite well and can be used in research concerning diverse decoding methods.

As expected, from the work of papers like *Comparison of Diverse Decoding Methods from Conditional Language Models*, when switching from beam search to diverse methods, there is a drop off in terms of summary quality measured by ROUGE score. Of the methods studied here the diversity/quality tradeoff is the greatest with nucleus sampling and the smallest with top-*k* combined with temperature.

4.6 Sample | Fine-tuned | $k = 640$, $k = 640$ and $t = 0.7$

Fine-tuned $k = 640$	Fine-tuned $k = 640$, $t = 0.7$
James Best played Sheriff Rosco in "The Dukes of Hazzard" TV show.<n>Best was a busy actor for decades wearing theater and in Hollywood.<n>"Give Uncle Jesse my love when you see him dear friend," co-star John Schneider tweets. Dr. Anthony Moschetto is accused of trying to hire informant and policeman to have rival Karpuss, 62,itis hurt or killed.<n>Police officers allegedly discovered approximately 100 weapons at Moschetto's home.	James Best died Monday of complications from pneumonia, friend says.<n>He was best known for his role as bumbling sheriff Rosco P. Coltrane on "The Dukes of Hazzard" Dr. Anthony Moschetto is accused of trying to have another physician hurt or killed.<n>His attorney says the allegations are "completely unsubstantiated"<n>Mosheto pleaded not guilty to all charges Wednesday.

A comparison like this shows how much temperature works to stabilize top-*k* decoding. The summaries moderated with temperature on the right are wholly free of the obvious errors and awkward phrasing underlined in the top-*k*-only summaries on the left.

4.7 Sample | Fine-tuned | $p = 0.80, 0.85, 0.90, 0.95$

$p = 0.80$	$p = 0.85$	$p = 0.90$	$p = 0.95$
Dr. Anthony Moschetto's attorney says his client is presumed to be innocent.<n>Moshett o is charged in what authorities say was a failed plot to have another physician hurt or killed.<n>Two other men have been named as accomplices, according to prosecutors. President Barack Obama says he gets letters from people asking him to change his mind on health care.<n>He spoke to Gupta about the science behind climate change and public health.<n>The President said the Supreme Court challenge to Obamacare is "the last gasp of folks... fighting against" it for ideological reasons.	Dr. Anthony Moschetto is a cardiologist on Long Island.<n>He is accused of trying to arrange the hurting and killing of another doctor.<n>One of his lawyers calls the allegations against him "completely unsubstantiated" In an interview with Sally Kohn, President Obama emphasizes the health benefits of climate change.<n>Obama stresses the scientific data in his climate change messages.	Dr. Anthony Moschetto is accused of trying to organise a hit on a fellow cardiologist.<n>Auth orities say he was trying to get his rival beaten and killed.<n>Moschot's lawyer says he will be cleared his client, who he says "is presumed to be innocent" President Obama urged Americans to make their voices heard about climate change.<n>John Sutter: "No challenge poses more of a public threat than climate change"	Dr. Anthony Moschetto, 55, is accused of plotting to have a rival doctor killed, authorities say.<n>His attorney sayslades are "completely unsubstantiated" and Moschetto is presumed innocent Cipriano. President Barack Obama under pressure from the bench to act on climate change sterility.<n>The President doesn't worry about Supreme Court ruling on health care.<n>"We can do something about it," Obama says.

As the extremely low ROUGE scores suggested, the resulting summaries from nucleus sampling with $p = 0.95$ are of extremely low quality. The underlined sentences feature bizarre word choices that make them ultimately nonsensical, and more generally, one sentence does not

flow well to the next. As the value of *p* decreases the summaries feature fewer, less severe mistakes.

5. Conclusion

This paper explored how fine-tuning could be used to deal with the loss in summary quality when moving from deterministic decoding methods like beam search to stochastic, diverse decoding methods like top-*k*, top-*k* with temperature, and nucleus sampling. Because the fine-tuned models start out at a much stronger point, yes they do ultimately return stronger ROUGE scores when using diverse decoding methods, but what is much more pertinent is that the results here argue against placing too much importance on ROUGE. Considering both the issues with ROUGE on a conceptual level and actual, generated summaries, it seems to be the case that, with an abstractive summarizer that is significantly sophisticated enough, like PEGASUS, somewhat low ROUGE scores do not necessarily mean poor summaries. But, this is not to say that ROUGE is totally meaningless, even for abstractive summarization. Indeed for several of the results here, low ROUGE score did, in fact, correlate with low summary quality.

Concerning how much the respective decoding methods impact ROUGE scores, the experiments found that top-*k* sampling with temperature makes the smallest sacrifice of quality for diversity, while nucleus sampling takes the greatest hit to quality in its pursuit of diversity. Finally, the results here support that measure of textual lexical diversity (MTLD) largely tracks with the expected results when using perplexity and is thus a viable alternative with which to investigate diversity in text.

Bibliography

Deutsch, Daniel, and Dan Roth. “Understanding the Extent to Which Summarization Evaluation Metrics Measure the Information Quality of Summaries.” *ArXiv:2010.12495 [Cs]*, Oct. 2020. *arXiv.org*, http://arxiv.org/abs/2010.12495.

Ganesan, Kavita. “ROUGE 2.0: Updated and Improved Measures for Evaluation of Summarization Tasks.” *ArXiv:1803.01937 [Cs]*, Mar. 2018. *arXiv.org*, http://arxiv.org/abs/1803.01937.

Holtzman, Ari, et al. “The Curious Case of Neural Text Degeneration.” *ArXiv:1904.09751 [Cs]*, Feb. 2020. *arXiv.org*, http://arxiv.org/abs/1904.09751.

Huang, Dandan, et al. “What Have We Achieved on Text Summarization?” *ArXiv:2010.04529 [Cs]*, Oct. 2020. *arXiv.org*, http://arxiv.org/abs/2010.04529.

Ippolito, Daphne, et al. “Comparison of Diverse Decoding Methods from Conditional Language Models.” *ArXiv:1906.06362 [Cs]*, June 2019. *arXiv.org*, http://arxiv.org/abs/1906.06362.

Lin, Chin-Yew. *ROUGE: A Package for Automatic Evaluation of Summaries*. ACL 2004, 2004.

McCarthy, Philip M., and Scott Jarvis. “MTLD, Vocd-D, and HD-D: A Validation Study of Sophisticated Approaches to Lexical Diversity Assessment.” *Behavior Research Methods*, vol. 42, no. 2, 2, May 2010, pp. 381–92. *DOI.org (Crossref)*, https://doi.org/10.3758/BRM.42.2.381.

Ng, Jun-Ping, and Viktoria Abrecht. “Better Summarization Evaluation with Word Embeddings for ROUGE.” *ArXiv:1508.06034 [Cs]*, Aug. 2015. *arXiv.org*, http://arxiv.org/abs/1508.06034.

See, Abigail, et al. “Get To The Point: Summarization with Pointer-Generator Networks.” *ArXiv:1704.04368 [Cs]*, Apr. 2017. *arXiv.org*, http://arxiv.org/abs/1704.04368.

Welleck, Sean, Ilia Kulikov, Jaedeok Kim, et al. “Consistency of a Recurrent Language Model With Respect to Incomplete Decoding.” *ArXiv:2002.02492 [Cs, Stat]*, Oct. 2020. *arXiv.org*, http://arxiv.org/abs/2002.02492.

Welleck, Sean, Ilia Kulikov, Stephen Roller, et al. “Neural Text Generation with Unlikelihood Training.” *ArXiv:1908.04319 [Cs, Stat]*, Sept. 2019. *arXiv.org*, http://arxiv.org/abs/1908.04319.

Zhang, Jingqing, et al. “PEGASUS: Pre-Training with Extracted Gap-Sentences for Abstractive Summarization.” *ArXiv:1912.08777 [Cs]*, July 2020. *arXiv.org*, http://arxiv.org/abs/1912.08777.