# Comparing Large Language Models for Paraphrasing

**Sam Malik**
srma2020@mymail.pomona.edu

## Abstract

As of recent, large language models, particularly a part of the Generative Pre-Trained series, have demonstrated themselves to be powerful text generation models. Models such as GPT-2 (Radford et al., 2018) reveal that large language models have strong zero-shot capabilities in a variety of downstream natural language processing tasks. Other models, built for sequence to sequence modeling, such as PEGASUS, and BART have profound text summarization capabilities which can be adapted to paraphrasing. In this paper, I present an effective method for adapting GPT-2 for paraphrasing, and compare its paraphrasing outputs to fine tuned BART and PEGASUS based models. Results show that GPT-2 based models produce less diverse paraphrases than PEGASUS and BART; GPT-2 based paraphrases do not alter lexical form as much as PEGASUS does.

## 1 Introduction

Paraphrase generation is a task in natural language processing (NLP) that is a sub domain of text summarization. It involves the transformation of a given sentence to a new sentence with different lexical form but the same semantic meaning (paperswithcode).

Some of the current open source sequence to sequence neural architectures available for paraphrasing are fine tuned versions of an abstractive text summarization model, PEGASUS (Zhang et al., 2019a), and an auto-regressive denoising sequence to sequence model, BART (Lewis et al., 2019). Although these models were not pre-trained for paraphrasing, their profound capabilities in text summarization and sequence to sequence modeling lend them prime candidates for paraphrasing models.

Generative Pretrained Transformer 2 (GPT-2), on the other hand, is a large language model that has the ability to perform well in zero shot settings. As GPT-2 is a language model, it was not strictly pre-trained in a supervised setting for a specific task. The language model is able to perform well in a plethora of zero-shot settings, implying that high-capacity language models can learn to perform various tasks without the explicit need for supervision (Radford et al., 2018). The large capacity of language models such as GPT-2 allows them to develop intricate representations of natural languages which can be used for much more than text generation.

To extend on the current range of popular aforementioned sequence to sequence models, and explore a domain of NLP that was not explored during the advent of GPT-2, the system I propose therefore utilizes GPT-2 to paraphrase natural language, and assess its capability of producing high quality paraphrases that change lexical form but maintain semantic meaning. I also compare GPT-2 based paraphrasing models to other fine tuned BART and PEGASUS models from Hugging-face.

## 2 Related Work

In 2016, Prakash et al. (2016) took the supposed first step toward a novel deep neural approach to paraphrase generation was using a stacked residual Long Short Term Memory (LSTM) network. Their neural network consisted of stacked LSTM networks with residual connections between for efficient training of these models (Prakash et al., 2016). They evaluated their model on three different datasets and presented a strong baseline for neural paraphrase generation. This paper builds on this by focusing on transformers based architectures such as GPT-2.

In 2019, Witteveen and Andrews (2019) advanced into paraphrasing with large language models, more recent work has involved using GPT-2 for paraphrase generation, similar to the goal of my research in this paper.Their fine tuning approach to utilizing GPT-2 for paraphrasing demonstrated strong capabilities of not only sentence level para-

phrasing, but paraphrasing over long spans of token sequences such as paragraphs without the need for truncation or segmenting the paragraphs into smaller sentences. It was found that the tuned GPT-2 model produced similar paraphrasing as the training examples with no conditional input (except for an input sentence followed by the specific identifying sequence) (Witteveen and Andrews, 2019). This work hopes to adopt similar strategies to create an initial GPT-2 paraphrase model. Nonetheless, this paper will attempt to compare the paraphrasing capability of different sized GPT-2 models with other models such as BART, and PEGASUS.

## 3 Methodology

In this section, I present how GPT-2 was fine-tuned for sentence level paraphrasing, and the metrics used to evaluate the effectiveness of the various paraphrasing language models. The objective task of the model is to produce paraphrases of an input sentence. First, I fine-tuned three different sized GPT-2 models for paraphrasing on paraphrase sentence pairs, then compare the outputs of the three fine-tuned GPT-2 models to two other fine-tuned BART and PEGASUS based models.

### 3.1 Large Language Modelling and GPT-2

The use of large language models has become nearly ubiquitous for the use of transfer learning and fine tuning for downstream NLP tasks Witteveen and Andrews (2019). While these models are initially trained in an unsupervised fashion, they can be adapted for other tasks by additional supervised training. One of such models is GPT-2, a large language model pre-trained for estimating a probability distribution of, given a sequence of token history, the possibilities of a subsequent next token. Four versions of the model exist, each with a different number of parameters: 117M, 345M, 762M, and 1.54B parameters respectively (Radford et al., 2018) . Each of the models were trained on WebText, 40GB of text data from 8 million links containing human curated/filtered content (Radford et al., 2018) . At the core of the language modeling approach, given an initial prompt, repeated, auto-regressive sampling of the model will generate text. For instance, given an initial prompt of the tokens $s_1, s_2, s_3$, sampling from the model once would yield a probability distribution over what the next token $s_4$ would be; sampling from that distribution would yield $s_1, s_2, s_3, s_4$. For further generations, that output would then be auto-regressively be fed back into the model as history for generating the next token $s_5$, and so on. Prior to repeated sampling from the model, the initial prompt can specify tasks and possible inputs related that task to be performed; natural language provides a flexible way to specify these initial prompts using various symbols or sentences.

### 3.2 Fine Tuning GPT2 for Paraphrasing

Fine-tuning language models has been a popular method of specializing them for downstream tasks. It involves performing an additional updating of the parameters of the model on task specific data. As GPT-2 is pre-trained, fine-tuning GPT-2 with more specific data would therefore only update the model parameters to enhances its paraphrasing capability.

This paper fine-tunes GPT-2 on a supervised dataset of paraphrased sentences. Each of these sentence pairs individually are fed into the model. These training sentence pairs are formatted specifically then fed into the model: <$s$> $S$ </$s$> > > > > <$p$> $P$ </$p$>, where $S$ and $P$ are paraphrases, and $>>>>$ is a special token for the model to learn is prompt to paraphrase the sentence that occurs before it Witteveen and Andrews (2019).

Once the model is fine-tuned on all sentence pairs in the dataset, we sample from the GPT-2 model using the initial prompt <$s$> $S$ </$s$> > > > > <$p$>. The model should learn that the following tokens should be $P$, an appropriate paraphrase to $S$, followed by the </$p$> token once the paraphrase is complete Witteveen and Andrews (2019). Our results show that all GPT-2 models effectively learn both generate a paraphrase and append the token </$p$> after it.

This paper will only fine-tune the smallest three versions of the GPT-2 provided model on HuggingFace due to hardware constraints. This includes "gpt2" (117M parameters), "gpt2-medium" (345M parameters) and "gpt2-large" (774M parameters) (Radford et al., 2018) .

The fine-tuning process will use default hyperparameters, and only run one epoch to avoid overfitting and promote generalization.

### 3.3 BART

In 2019, Lewis et al. (2019) proposed BART. According to the paper, BART is a "denoising autoencoder for pretraining sequence-to-sequence models" (Lewis et al., 2019). The model utilizes a sequence to sequence encoder decoder architecture

for machine translation. It works well in settings where it is fine-tuned for comprehension, question answering, text generation, and summarization tasks (Lewis et al., 2019). As such, transfer learning on a BART model for paraphrasing would likely yield similarly good results. The paraphrasing model this will use in this paper is "eugenesiow/bart-paraphrase" from HuggingFace. The model was fine-tuned on BART for paraphrasing using paraphrase pairs from the Quora question pairs, Google PAWS and the MSR paraphrase corpus.

### 3.4 PEGASUS

Also in 2019, Zhang et al. (2019a) introduced "Pre-training with Extracted Gap-sentences for Abstractive Summarization" (PEGASUS). PEGASUS is a large transformer based encoder decoder model trained on generating masked sentences form a large corpus to an extractive summary (Zhang et al., 2019a). At the time, the paper achieved state of the art on 12 out of 12 summarization tasks it was tested on. This paper will use "tuner007/pegasus_paraphrase" from HuggingFace, a PEGASUS based model fine-tuned for paraphrasing.

### 3.5 Evaluation Metrics

To evaluate a model's paraphrases, this papers uses the Universal Sentence Encoder (USE) and ROUGE-L scores.

Given a fine-tuned model, we sample 10 paraphrases for a given input sentence. We then assess the quality of the paraphrases using popular metrics for measuring sentence translation quality and similarity, USE (Cer et al., 2018) and ROUGE-L (Lin, 2004).

The Universal Sentence Encoder (USE) is a model that encodes a sentence into a semantic embedding space of 512 dimensions (Cer et al., 2018). The similarity between an input sentence, $S$, and the paraphrases sentence from the model, $P$, is the cosine similarity between the USE generated sentence embeddings for $S$ and $P$. The range of this value is from -1 to 1

The ROUGE (recall oriented understudy for gisting evaluation) is a metric for evaluating summarization and machine translation by comparing a target and a reference sentence (Lin, 2004). The ROUGE-L score is divided into a precision and recall score. The ROUGE-L precision score between a target $T$ and a reference $R$ is the longest common subsequence between $T$ and $R$ divided by the number of unigrams in $T$. The ROUGE-L recall score is the longest common subsequence between $T$ and $R$ divided by the number of unigrams in $R$ (Chiusano). This papers uses the ROUGE-L F1 score, which is the equal harmonic mean of the ROUGE-L precision and recall scores. The range of this value is from 0 to 1.

Though paraphrase quality will vary across the 10 samples, the best quality paraphrase is one that has a high USE similarity score and a low ROUGE-L score because this represents high semantic similarity even with altered lexical form. If the ROUGE-L is too high, the input and paraphrase share too many words in common. Hence, to strike a balance between semantic similarity and lexical and syntactic difference, we also calculate the score $USE \times (1 - ROUGE_L)$ for each paraphrase as a balanced measure of the quality of a paraphrase.

## 4 Datasets

To finetune each of the GPT-2 models, I utilized the training split of three parallel sentence paraphrasing datasets on HuggingFace: Google PAWS, Quora Question Pairs, and TaPaCo (en).

### 4.1 Google PAWS

Google PAWS is a sentence level paraphrase data that contains 108,463 human-labeled data that exhibit context and word order information for paraphrase identification (Zhang et al., 2019b) . Models not trained on PAWS have mediocre results on the dataset. However, including the PAWS dataset in training improved accuracy up to 85% while retaining the performance on existing data. The data on HuggingFace was split into training, testing, and validation. The training of the GPT-2 models use the training portion of this data.

### 4.2 Quora

The Quora question pairs dataset is composed of parallel question pairs that are either paraphrases or not (HuggingFace). The original task of this dataset is to determine whether two questions are paraphrases. The dataset utilized the marker "is_duplicate" to indicate whether a training instance containing a sentence pair were paraphrases. In the training of the models in this paper, I utilized the portion of the data where "is_duplicate" was True.

### 4.3 TaPaco

TaPaCo is a multilingual paraphrasing corpus for 73 languages. Thecorupus contains a total of 1.9 million sentences, with approximately 200 - 250,000 sentences per language (Scherrer, 2020). The training of the models utilized the english portion of this dataset. This dataset was available on HuggingFace. I use the entire english dataset for training, and therefore do not use any in the testing data.

## 5 Results

This section will outline the results of the five paraphrasing models. It will compare the paraphrasing abilities of a fine tuned GPT2, GPT2-medium, GPT2-large, PEGASUS, and BART models.

Using the "test" portion on the Google PAWS dataset, and the "is_duplicate == False" portion of the Quora Dataset, I gathered 10 sentences of sentence lengths from 7 to 20, for a total of 140 sentences for validation. I allowed each model to generate 10 paraphrases for each of the 140 sentences, then calculated the USE similarity and ROUGE-L score for each paraphrase. I also calculate the $USE \times (1 - ROUGE_l)$ metric for each paraphrase. Then, for each model, I averaged the USE similarity, ROUGE-L, and composite scores for each sentence length. Comparing the paraphrases' USE, ROUGE-L, and the composite scores allow for robust comparisons between the paraphrasing abilities of each model.

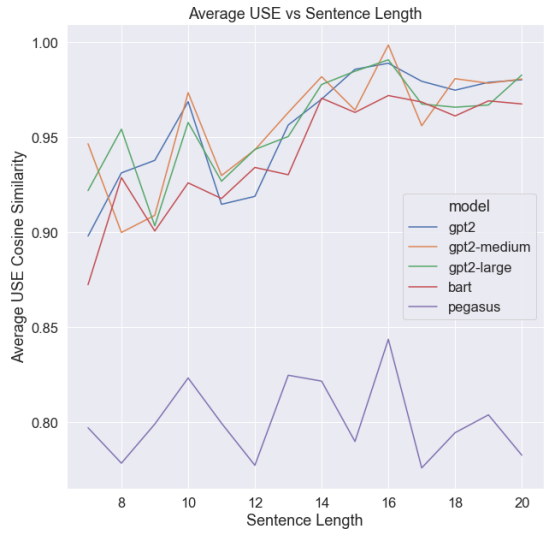


Figure 1: Average USE similarity score for a given sentence length for each model

Figure 1 shows the relationship between the average USE similarity score and the sentence length for each model. Here, there is a clear distinction between the PEGASUS based model and the other models. The GPT-2 models and the BART based model all see similar upward trends in the average USE similarity as the sentence length increases, with BART's line is slightly lower. The similarity between their inputs and paraphrases approach near perfect similarity. The PEGASUS model exhibits USE scores ranging from 0.778 to 0.847, significantly lower than the other models. However, this range of similarity scores implies that the paraphrase retains most of the semantic information from the input sentence.

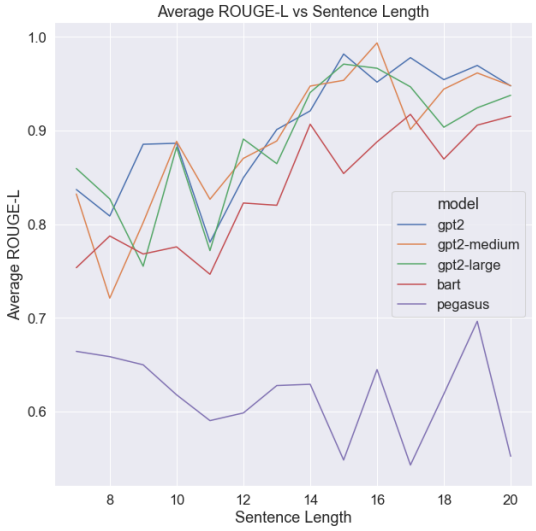


Figure 2: Average ROUGE-L score for a given sentence length for each model

In Figure 2, we see similar trends, where the GPT2 models and the BART based model exhibit increases in average ROUGE-L scores as the sentence length increases. The GPT-2 models increase from approximaely 0.8 and cross 0.9 at 13 worded sentences. The BART based model, stays below the GPT-2 models and only barely crosses a ROUGE-L of 0.9 at 14, 17, 19 and 20 worded sentences. PEGASUS remains below 0.7 across all sentence length.

A high ROUGE-L score suggests that the GPT-2 and the BART based models (GPT-2 more so than the BART model) are only changing the input sentence at the word level using synonyms or other close replacements rather than altering its lexical structure.

A large ROUGE-L between an input and a paraphrase implies that the paraphrase may still remain many of the same words as the input in the same

order. When sentences share the same words in the same order, the semantic meaning of the sentences would be much higher. Therefore, the large USE scores of the GPT-2 and BART models may be from replacing a few words with synonyms without changing the structure of the sentence, or not paraphrasing the input sentence at all.

Figure 2 also illustrates that the ROUGE-L scores for the PEGASUS based model are also lower than the other models, which suggests that PEGASUS is better at diversifying the paraphrase sentence such that it is not lexically close to the input sentence.

Out of the models, PEGASUS and BART consistently have the lower ROUGE-L scores than their GPT2 counterparts, which emphasizes that these fine tuned models are better at producing alternate forms while retaining semantic meaning.

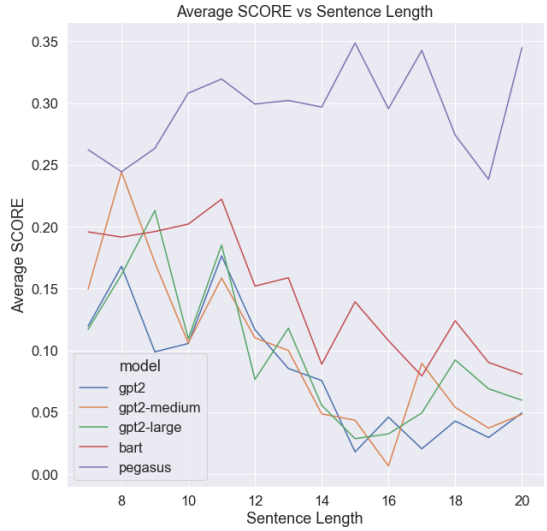


Figure 3: Average ROUGE-L score for a given sentence length for each model

When we average the $USE \times (1 - ROUGE_l)$ scores to score each model on their ability to produce semantically similar sentences while significantly altering the lexical form, we see that the PEGASUS based model outperforms every other model at every sentence length. PEGASUS is followed by the BART based model, followed by "gpt2-large", "gpt2-medium", "gpt2" for sentences longer than 16 words.

Additionally, all models apart from PEGASUS have decreasing composite scores as sentence length increases, which implies that longer input sentences and paraphrases decreases their ability to produce semantically similar sentences with altered lexical form. The PEGASUS based model consistently outperforms the other models on this metric, which implies that it has the ability to do this.

### 5.1 Paraphrase Examples

In Table 1, the GPT-2 models all produced identical paraphrases that exactly replicated the input sentence: both the USE and ROUGE-L score were equal to 1. BART's and PEGASUS' top three paraphrases both showed a deeper understanding of the input sentence by referencing MBA as a "business degree" or "masters degree", "B.Tech" as an entity related to engineering, or "job" as something related to earning money. In these examples, though, the lexical form deviated so much the paraphrases do not have the same semantics as the input.

In Table 2, the GPT-2 models and BART all produced similar paraphrases. In their paraphrases, dates were reformatted to represent month-day-year from day-month-year and the ordering of "North America and Europe" and the dates were swapped. In addition, the medium and large GPT-2 models and the BART model were able to replace the word "published" with a viable synonym "released." The PEGASUS based model, however, paraphrased the input to "The game was published in North America, Europe, and Japan in October and June of 2015," removing one of the dates in the sentence. This alteration completely changes the semantic meaning (and factual nature) of the sentence. When a paraphrase should involve only minor rearrangements to the lexical form of a sentence, the data suggests that the GPT-2 models outperform PEGASUS.

Lastly, in Table 3, the GPT-2 models were able to identify that this statement was a question, and correctly changed the punctuation at the end of the sentence to a question mark. Additionally, the small and medium GPT-2 models only changed the word "kinds" to the word "kind": the paraphrase is valid, and the sentence similarity is high, but the models did not change much of the lexical form of the sentence. PEGASUS and BART had the most diversity in responses, and were both able to remove words from the input sentence such that the final paraphrase still retains similar semantic meaning to the first, for instance, the paraphrase "What words would you use to describe jewelry?"

| Input: "MBA after B.TECH or JOB in IT." | USE | ROUGE-L | Composite |
|---|---|---|---|
| gpt2 | | | |
| "MBA after B.TECH or JOB in IT." | 1 | 1 | 0 |
| "MBA after B.TECH or JOB in IT." | 1 | 1 | 0 |
| "MBA after B.TECH or JOB in IT." | 1 | 1 | 0 |
| gpt2-medium | | | |
| "MBA after B.TECH or JOB in IT." | 1 | 1 | 0 |
| "MBA after B.TECH or JOB in IT." | 1 | 1 | 0 |
| "MBA after B.TECH or JOB in IT." | 1 | 1 | 0 |
| gpt2-large | | | |
| "MBA after B.TECH or JOB in IT?" | 1 | 1 | 0 |
| "MBA after B.TECH or JOB in IT?" | 1 | 1 | 0 |
| "MBA after B.TECH or JOB in IT?" | 1 | 1 | 0 |
| BART | | | |
| 'What is the best field for earning money after completing a B.Tech in IT engineering?' | 0.651 | 0.416 | 0.380 |
| 'What is the best field for earning money after completing a B.Tech in IT?' | 0.649 | 0.434 | 0.367 |
| 'What is the best field for earning money after completing a B.Tech in computer science?' | 0.636 | 0.33 | 0.426 |
| PEGASUS | | | |
| 'After B.TECH or a job in IT, you can get a MBA.' | 0.915 | 0.66 | 0.305 |
| 'After B.TECH or a job in IT, you can get a business degree.' | 0.779 | 0.636 | 0.283 |
| 'After B.TECH or a job in IT, you can go for a masters degree.' | 0.776 | 0.608 | 0.303 |

Table 1: Paraphrases for "MBA after B.TECH or JOB in IT."

| Input: 'The game was published on 2 October 2014 in North America and Europe and in Japan in June 2015 .' | USE | ROUGE-L | Composite |
|---|---|---|---|
| gpt2 | | | |
| 'The game was published in North America and Europe on October 2, 2014 and in Japan on June 15, 2015.' | 0.936 | 0.717 | 0.264 |
| 'The game was published in North America and Europe on October 2, 2014 and in Japan on June 15, 2015.' | 0.936 | 0.717 | 0.264 |
| 'The game was published in North America and Europe on October 2, 2014 and in Japan on June 15, 2015.' | 0.936 | 0.717 | 0.264 |
| gpt2-medium | | | |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan on June 15, 2015.' | 0.903 | 0.666 | 0.301 |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan on June 15, 2015.' | 0.903 | 0.666 | 0.301 |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan on June 15, 2015.' | 0.903 | 0.666 | 0.301 |
| gpt2-large | | | |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan in June 2015.' | 0.951 | 0.736 | 0.250 |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan in June 2015.' | 0.951 | 0.736 | 0.250 |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan in June 2015.' | 0.951 | 0.736 | 0.250 |
| BART | | | |
| 'The game was released in North America and Europe on 2 October 2014 and in Japan in June 2015.' | 0.960 | 0.736 | 0.252 |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan in June 2015.' | 0.951 | 0.736 | 0.250 |
| 'The game was released in North America and Europe on October 2, 2014, and in Japan in June 2015.' | 0.951 | 0.736 | 0.250 |
| PEGASUS | | | |
| 'The game was published in North America, Europe, and Japan in October and June of 2015.' | 0.856 | 0.742 | 0.220 |
| 'The game was released in North America, Europe, and Japan in October and June of 2015.' | 0.833 | 0.685 | 0.261 |
| 'The game was released in North America, Europe, and Japan in October and June of 2015.' | 0.833 | 0.685 | 0.261 |

Table 2: Paraphrases for 'The game was published on 2 October 2014 in North America and Europe and in Japan in June 2015 .'

| Input: 'What kinds of words would you use to describe jewelry.' | USE | ROUGE-L | Composite |
|---|---|---|---|
| gpt2 | | | |
| 'What kind of words would you use to describe jewelry?' | 0.985 | 0.9 | 0.098 |
| 'What kind of words would you use to describe jewelry?' | 0.985 | 0.9 | 0.098 |
| 'What kind of words would you use to describe jewelry?' | 0.985 | 0.9 | 0.098 |
| gpt2-medium | | | |
| 'What kind of words would you use to describe a piece of jewelry?' | 0.945 | 0.782 | 0.205 |
| 'What kind of words would you use to describe a piece of jewelry?' | 0.945 | 0.782 | 0.205 |
| 'What kind of words would you use to describe a piece of jewelry?' | 0.945 | 0.782 | 0.205 |
| gpt2-large | | | |
| 'What kind of words would you use to describe a necklace?' | 0.893 | 0.761 | 0.212 |
| 'What kind of words would you use to describe a necklace?' | 0.893 | 0.761 | 0.212 |
| 'What kind of words would you use to describe a necklace?' | 0.893 | 0.761 | 0.212 |
| BART | | | |
| 'What kinds of words would you use to describe jewelry?' | 1.0 | 1.0 | 0.0 |
| 'What kind of words would you use to describe jewelry?' | 0.985 | 0.9 | 0.098 |
| 'What words would you use to describe jewelry?' | 0.952 | 0.888 | 0.105 |
| PEGASUS | | | |
| 'What types of words would you use to describe jewelry?' | 0.990 | 0.9 | 0.099 |
| 'What kind of words would you use to describe jewelry?' | 0.985 | 0.9 | 0.098 |
| 'What words would you use to describe jewelry?' | 0.952 | 0.888 | 0.105 |

Table 3: Paraphrases for "What kinds of words would you use to describe jewelry."

## 6 Conclusion

This paper finetuned three GPT-2 paraphrase models of different parameter sizes and compare its output to a BART based and a PEGASUS based paraphrasing model. While GPT-2 produced valid paraphrases that retained semantic meaning of the input sentence, the other models, PEGASUS and BART, managed to produce more robust and diversified paraphrases: ones that altered lexical form yet retained semantic meaning. GPT-2 paraphrases seemed to perform better on paraphrasing input sentences that do not rely on much lexical altering.

## 7 Acknowledgments

The original plan for this paper was to finetune a GPT-2 paraphraser with controlled sentiment. The research project would ask the question, "To what extent are large language models such as GPT-2 effective or controlled sentiment parahrasing." finetuning GPT-2 to paraphrase a sentence into a specific sentiment would have required Proximal Policy Optimization reinforcement learning, which is a very computationally expensive and resource intensive. Due to the hardware constraints (GPU RAM not sufficient enough on Google Colab Pro), I had to forgo the controlled sentiment aspect of the project. After this, I decided to investigate other paraphrasing models and conduct research in the comparison between them.

## References

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Fabio Chiusano. Two minutes nlp — learn the rouge metric by examples.

HuggingFace. Datasets: quora.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

paperswithcode. Paraphrase generation.

Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2016. Neural paraphrase generation with stacked residual lstm networks.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Yves Scherrer. 2020. TaPaCo: A corpus of sentential paraphrases for 73 languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6868–6873, Marseille, France. European Language Resources Association.

Sam Witteveen and Martin Andrews. 2019. Paraphrasing with large language models. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019b. Paws: Paraphrase adversaries from word scrambling.