Polly Want a Cracker: Analyzing Performance of Parroting on Paraphrase Generation Datasets

Hong-Ren Mao, Hung-Yi Lee
Department of Electrical Engineering
National Taiwan University
ccm29cam@gmail.com, hungyilee@ntu.edu.tw

Abstract

Paraphrase generation is an interesting and challenging NLP task which has numerous practical applications. In this paper, we analyze datasets commonly used for paraphrase generation research, and show that simply parroting input sentences surpasses state-of-the-art models in the literature when evaluated on standard metrics. Our findings illustrate that a model could be seemingly adept at generating paraphrases, despite only making trivial changes to the input sentence or even none at all.

1 Introduction

The task of paraphrase generation has many important applications in NLP. It can be used to generate adversarial examples of input text, which can then be used to train neural networks so that they become less susceptible to adversarial attack (Iyyer et al., 2018). For knowledge-based QA systems, a paraphrasing step can produce multiple variations of a user query and match them with knowledge base assertions, enhancing recall (Yin et al., 2015; Fader et al., 2014). Relation extraction can also benefit from incorporating paraphrase generation into its processing pipeline (Romano et al., 2006). Manually annotating translation references is expensive, and automatically generating references through paraphrasing has been shown to be effective for evaluation of machine translation (Zhou et al., 2006; Kauchak and Barzilay, 2006).

Datasets used for paraphrase generation include QUORA[1], TWITTER (Lan et al., 2017) and MSCOCO (Lin et al., 2014). Previous work on paraphrase generation that used these datasets (Wang et al., 2019; Gupta et al., 2018; Li et al., 2018; Prakash et al., 2016) chose BLEU (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007) and TER (Snover et al., 2006) as evaluation metrics.

In this paper, we find that simply using the input sentence as output in an unsupervised manner (*i.e.* fully parroting the input) significantly outperforms the state-of-the-art on two metrics for TWITTER, and on one metric for QUORA. Even after changing part of the input sentence (*i.e.* partially parroting the input), state-of-the-art metric scores can still be surpassed.

Consequently, for future paraphrase generation research which achieve good evaluation scores, we suggest investigating whether their methods or models act differently from simple parroting behavior.

2 Method Description

Given an input sentence i, the goal of paraphrase generation is to generate an output sentence o which is semantically identical to i, but contain variations in lexicon or syntax. Full parroting simply uses the input as output ($o = i$).

Paraphrase generation models may not parrot the input sentence word for word, but it is possible that they only modify a few words of the input, thus we also experiment with simple methods of modifying i, such as replacing or cutting words from the head, from the tail or from random positions.

Both full parroting and the forms of partial parroting we use are fully unsupervised.

3 Datasets

QUORA. The QUORA dataset contains 149,263 paraphrase sentence pairs (positive examples) and 255,027 non-paraphrase sentence pairs (negative examples). Having both positive and negative ex-

[1] https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 5960–5968,
Hong Kong, China, November 3–7, 2019. ©2019 Association for Computational Linguistics

amples makes it appealing for research on paraphrase generation (Gupta et al., 2018; Li et al., 2018) and identification (Lan and Xu, 2018). After processing the dataset, there are 149,650 unique sentences that have reference paraphrases.

Gupta et al. (2018) sampled 4K sentences as their test set, but did not specify which sentences they used. Li et al.(2018) sampled 30K sentences as their test set, also not specifying which sentences they used. To avoid selecting a subset of data that is biased in favor of our method, we perform evaluation on the entire QUORA dataset. Although we evaluate on the entire dataset, the size of our training set is zero due to the fully unsupervised nature of full and partial parroting.

We group sentences by the number of reference paraphrases they have, and plot the relative counts in Appendix A. It can be seen that over 64% of entries have only a single reference paraphrase, which is problematic because even if a paraphrase of good quality is generated for any one of these entries, BLEU, METEOR and TER scores could still be inferior if the generated paraphrase differs too much from the single reference paraphrase. Previous paraphrase generation work on QUORA (Gupta et al., 2018; Li et al., 2018) did not mention removing these entries, thus we include them in our experiments for fair comparison. However, we strongly recommend future work which wishes to use BLEU, METEOR and TER as evaluation metrics to only consider entries that have multiple reference paraphrases.

TWITTER. There are 114,025 paraphrase sentence pairs in TWITTER, which were acquired by collecting tweets which contain identical URLs (Lan et al., 2017). As with QUORA, prior paraphrase generation work on this dataset (Li et al., 2018) did not provide their sampled test set sentences, so we evaluate parroting on the entire dataset to avoid bias. We follow the same data processing steps as QUORA, and plot the number of reference paraphrases in Appendix A.

MSCOCO. This is an image captioning dataset, with multiple captions provided for a single image (Lin et al., 2014). There have been multiple works which use it as a paraphrase generation dataset by treating captions of the same image as paraphrases (Wang et al., 2019; Gupta et al., 2018; Prakash et al., 2016). The training and testing sets are available, containing 331,163 and 162,016 input sentences respectively.

However, relevance scores for captions of the same image score only 3.38 out of 5 under human evaluation (in contrast, the score is 4.82 for QUORA) (Gupta et al., 2018), due to the fact that different captions for the same image often vary in the semantic information conveyed. This makes the use of MSCOCO as a paraphrase generation dataset questionable.

We plot the number of reference paraphrases in Appendix A.

4 Experiments

We evaluate the performance of full parroting on all three datasets and compare with state-of-the-art models.

We also study the performance of partial parroting. Whereas full parroting does not modify the input sentence, partial parroting replaces or cuts some of the input words. We try three different modes of choosing words to be cut or replaced: from the sentence head, from the tail or sampled randomly.

4.1 Evaluation

Following prior paraphrase generation research which used QUORA, TWITTER and MSCOCO, we use BLEU, METEOR and TER as evaluation metrics. When calculating metric scores, all available reference paraphrases for a given input sentence are considered.

4.2 Results

Full parroting. Our results are organized in Tables 1, 2 and 3. We see for TWITTER, parroting outperforms the state-of-the-art by significant margins on both BLEU and METEOR scores; for QUORA, parroting outperforms the state-of-the-art appreciably on METEOR while having comparable performance on BLEU.

The poor performance of full parroting on MSCOCO is due to higher edit distances between input sentences and their reference paraphrases. TER measures the edit distance of a sentence to a reference sentence, normalized by the average length of all references (Snover et al., 2006):

$$\text{TER} = \frac{\text{\# of edits}}{\text{average \# of reference words}}$$

We see that the TER score of full parroting is particularly high on MSCOCO compared to the other two datasets. Correspondingly, the BLEU and METEOR scores are lower by a wide margin.

Metric	STATE-OF-THE-ART			PARROT		
	paper	score	num_train	score	num_train	ΔSOTA
BLEU ↑	(Li et al., 2018)	**43.54**	100K	41.59	0	-4.47%
METEOR ↑	(Gupta et al., 2018)	33.6	150K	**38.60**	0	**+14.88%**
TER ↓	(Gupta et al., 2018)	**39.5**	150K	45.22	0	+14.47%

Table 1: Performance of full parroting v.s. state-of-the-art on QUORA. Higher BLEU and METEOR scores are better, while higher TER scores are worse. Bold text represents best results.

Metric	STATE-OF-THE-ART			PARROT		
	paper	score	num_train	score	num_train	ΔSOTA
BLEU ↑	(Li et al., 2018)	45.74	110K	**65.26**	0	**+42.67%**
METEOR ↑	(Li et al., 2018)	20.18	110K	**41.73**	0	**+106.77%**
TER ↓	-	-	-	**41.87**	0	-

Table 2: Performance of full parroting v.s. state-of-the-art on TWITTER.

Metric	STATE-OF-THE-ART			PARROT		
	paper	score	num_train	score	num_train	ΔSOTA
BLEU ↑	(Wang et al., 2019)	**44.0**	331K	19.0	0	-56.8%
METEOR ↑	(Wang et al., 2019)	**34.7**	331K	23.9	0	-31.0%
TER ↓	(Wang et al., 2019)	**37.1**	331K	70.4	0	+89.7%

Table 3: Performance of full parroting v.s. state-of-the-art on MSCOCO.

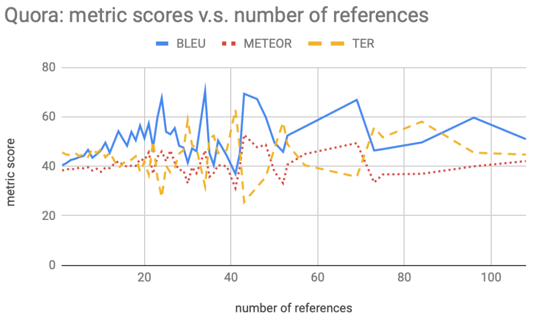

Figure 1: Metric scores v.s. number of reference paraphrases (Quora). For lower numbers of references, metric scores improve as the amount of references increases. For higher numbers of references, there does not appear to be a clear correlation.

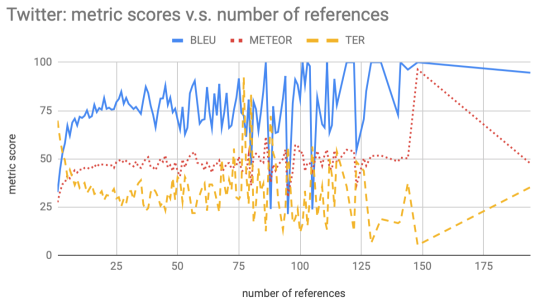

Figure 2: Metric scores v.s. number of reference paraphrases (Twitter). For lower numbers of references, metric scores improve as the amount of references increases. For higher numbers of references, there does not appear to be a clear correlation.

For further investigation of parroting on QUORA and TWITTER, we plot parroting performance versus the number of reference paraphrases available for a given input sentence (Figures 1 and 2). If the number of references is not too high, metric scores generally improve when the number of references rises. Once the number of references exceeds a certain threshold, we do not observe a clear correlation, showing that the probability of finding a reference sentence which bears higher resemblance to the input does not increase proportionally with the number of references.

The choice of testing on the entire dataset for QUORA and TWITTER experiments was to avoid bias in favor of parroting. Nevertheless, we also randomly sampled test sets of size 4K for QUORA in the same manner as (Gupta et al., 2018) (which holds the most state-of-the-art records on QUORA) and test sets of size 5K for TWITTER in the same manner as (Li et al., 2018) (which holds all state-

Statistic	QUORA (5K test set × 1200)			TWITTER (4K test set × 250)		
	BLEU ↑	METEOR ↑	TER ↓	BLEU ↑	METEOR ↑	TER ↓
Average	41.57	38.59	45.21	74.97	46.22	33.43
Std. Dev.	0.50	0.20	0.50	0.36	0.14	0.35
Max.	43.12	39.29	46.95	76.01	46.56	34.31
Min.	39.98	37.85	43.53	74.01	45.89	32.41

Table 4: Performance of full parroting on randomly sampled test sets. The test set size and sampling method is the same as that described in prior state-of-the-art work. Here, scores for sampled QUORA test sets are similar to those of the full dataset, and the scores for TWITTER test sets are better than scores achieved on the full dataset.

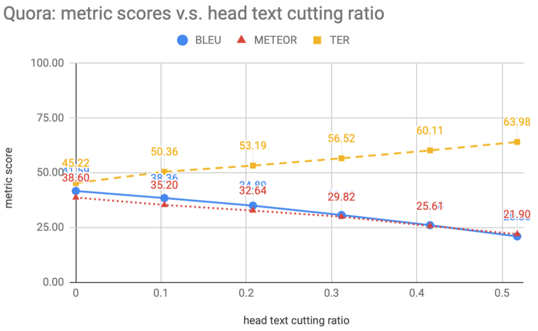

Figure 3: Metric scores v.s. ratio of text that is cut from the start of the input sentence (Quora).

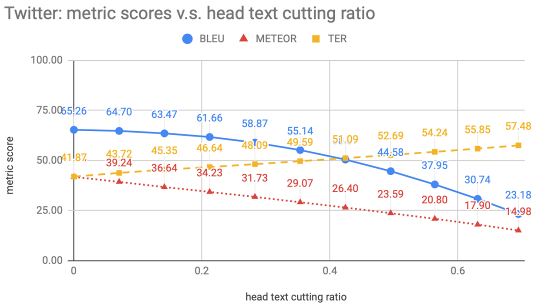

Figure 4: Metric scores v.s. ratio of text that is cut from the start of the input sentence (Twitter).

of-the-art records on TWITTER). In total, 1200 test sets of size 4K were sampled for QUORA and 250 test sets of size 5K were sampled for TWITTER. Parroting performance on these sampled test sets can be found in Table 4. It can be observed that the average metric scores for QUORA are similar to the scores in Table 1, whereas the average scores for TWITTER are noticeably better than those in Table 2. Furthermore, the score deviation between different samples is small. Consequently, although the exact test sets used by (Gupta et al., 2018) and (Li et al., 2018) are not available, it is logical to assume that parroting performance would still exceed or be on par with the state-of-the-art on those test sets.

Partial parroting. We also introduce lexical variation into our parroting method by replacing or cutting words of the input sentence. For replacement, we substitute input words with an out-of-vocabulary word not found in any of the input sentence's reference paraphrases. Paraphrase generation models are usually allowed to generate words which exist in reference paraphrases; we purposely use out-of-vocabulary words to give harsher scores to our method.

Figures 3 and 4 show performance of cutting words from the start of input sentences. For QUORA, when over 10% of the input sentence has been modified by being cut off, partial parroting underperforms the state-of-the-art by only 3.8% on METEOR. For TWITTER, the same form of partial parroting (cutting off words) still outperforms the state-of-the-art on BLEU when input sentences are modified by 42% , and does the same on METEOR when the input is modified by 56%. Additionally, we experiment with cutting words in other positions, and also replace words rather than cut them away. The results can be found in Appendix B.

Earlier work using QUORA and TWITTER (Gupta et al., 2018; Li et al., 2018) only provided a few examples of output paraphrases, and did not study in detail the paraphrasing behavior of their models, making it unclear whether the models achieve qualitatively better results than our simple rule-based parroting techniques, given that evaluation scores of the two are similar. We recommend future research to perform such an analysis if their metric scores are close to that of parroting.

5 Related Work

For the task of paraphrase generation, Wang et al. (2019) trained a Transformer network on

MSCOCO; Gupta et al. (2018) trained a seq2seq variational autoencoder (VAE) on QUORA and MSCOCO; Li et al. (2018) trained a seq2seq pointer network on QUORA and TWITTER, then fine-tuned it with an *evaluator* which was trained via inverse reinforcement learning; Prakash et al. (2016) trained a seq2seq model with residual connections on MSCOCO.

Work on paraphrase generation using other datasets can also be found. Methods include lexical substitution (Hassan et al., 2007; Bolshakov and Gelbukh, 2004), back-translation (Wieting and Gimpel, 2018) and sequence-to-sequence neural networks (Iyyer et al., 2018).

It is worth noting that paraphrase generation serves practical purposes, such as augmenting training data for NLP models to decrease their susceptibility to adversarial attack (Iyyer et al., 2018), or enhancing recall for QA systems (Yin et al., 2015; Fader et al., 2014). Improvement of downstream model performance is a valid evaluation metric for paraphrase generation, and future work wishing to use QUORA entries which only have a single reference paraphrase could choose such an evaluation metric instead of BLEU, METEOR or TER.

As a sidenote, we also ran experiments in which BLEU scores were calculated using non-reference dataset sentences. The results are in Appendix C.

6 Conclusion

In this work, we discover that various forms of simple parroting outperforms state-of-the-art results on QUORA and TWITTER when evaluated using BLEU and METEOR. An interpretation is that current models could simply be parroting input sentences, and researchers should perform qualitative analysis of such behavior. Another interpretation is that BLEU and METEOR are inappropriate for evaluating paraphrase generation models, in which case other metrics such as effectiveness of data augmentation (Iyyer et al., 2018), may be used instead.

References

Igor A. Bolshakov and Alexander Gelbukh. 2004. Synonymous paraphrasing using wordnet and internet. In *Natural Language Processing and Information Systems*, pages 312–323, Berlin, Heidelberg. Springer Berlin Heidelberg.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1156–1165, New York, NY, USA. ACM.

Ankush Gupta, Arvind Agarwal, Prawaan Singh, and Piyush Rai. 2018. A deep generative framework for paraphrase generation. In *AAAI*.

Samer Hassan, Andras Csomai, Carmen Banea, Ravi Sinha, and Rada Mihalcea. 2007. Unt: Subfinder: Combining knowledge sources for automatic lexical substitution. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 410–413. Association for Computational Linguistics.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885. Association for Computational Linguistics.

David Kauchak and Regina Barzilay. 2006. Paraphrasing for automatic evaluation. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 455–462, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *EMNLP*.

Wuwei Lan and Wei Xu. 2018. Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zichao Li, Xin Jiang, Lifeng Shang, and Hang Li. 2018. Paraphrase generation with deep reinforcement learning. In *EMNLP*.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

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. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2923–2934. The COLING 2016 Organizing Committee.

Lorenza Romano, Milen Kouylekov, Idan Szpektor, Ido Dagan, and Alberto Lavelli. 2006. Investigating a generic paraphrase-based approach for relation extraction. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231.

Su Wang, Rahul Gupta, Nancy Chang, and Jason Baldridge. 2019. A task in a suit and a tie: paraphrase generation with semantic augmentation. In *AAAI*.

John Wieting and Kevin Gimpel. 2018. Paranmt-50m: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462. Association for Computational Linguistics.

Pengcheng Yin, Nan Duan, Ben Kao, Junwei Bao, and Ming Zhou. 2015. Answering questions with complex semantic constraints on open knowledge bases. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 1301–1310, New York, NY, USA. ACM.

Liang Zhou, Chin-Yew Lin, and Eduard H. Hovy. 2006. Re-evaluating machine translation results with paraphrase support. In *EMNLP*.

A Number of References Paraphrases for Datasets

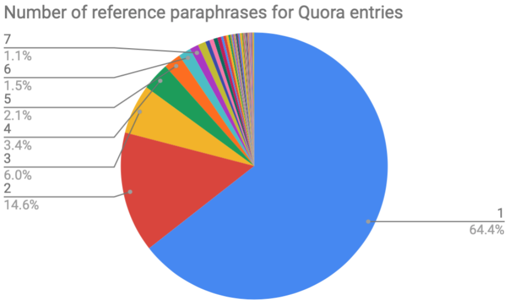

Figure 5: Number of reference paraphrases v.s. percentage of Quora dataset

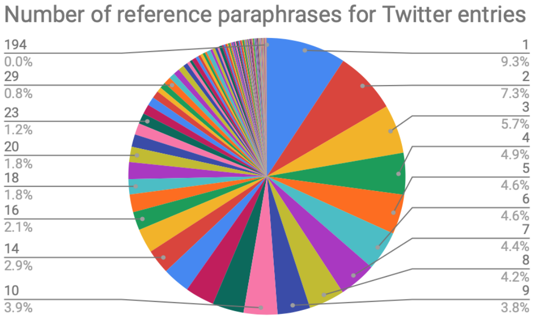

Figure 6: Number of reference paraphrases v.s. percentage of Twitter dataset

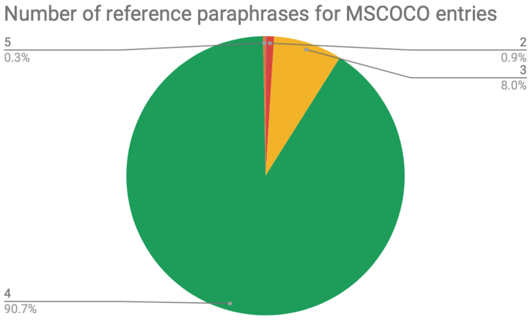

Figure 7: Number of reference paraphrases v.s. percentage of MSCOCO dataset

B Performance of Partial Parroting

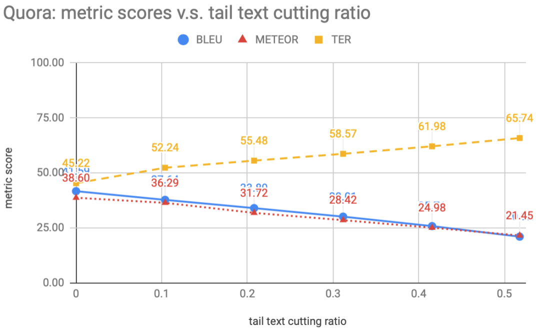

Figure 8: Metric scores v.s. ratio of text that is cut from the end of the input sentence (Quora)

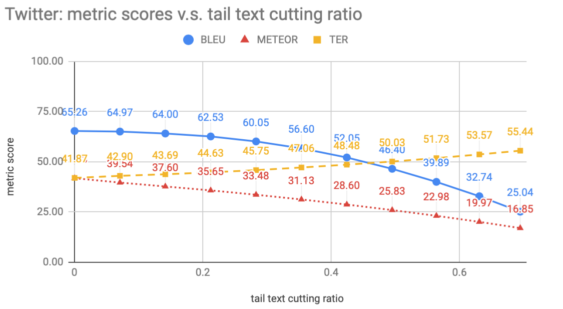

Figure 9: Metric scores v.s. ratio of text that is cut from the end of the input sentence (Twitter)

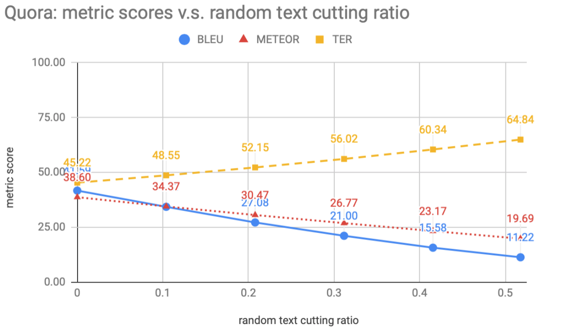

Figure 10: Metric scores v.s. ratio of text that is cut randomly from the input sentence (Quora)

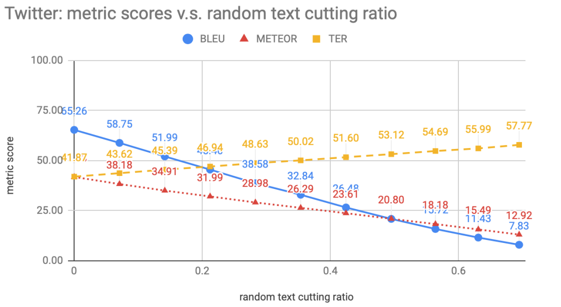

Figure 11: Metric scores v.s. ratio of text that is cut randomly from the input sentence (Twitter)

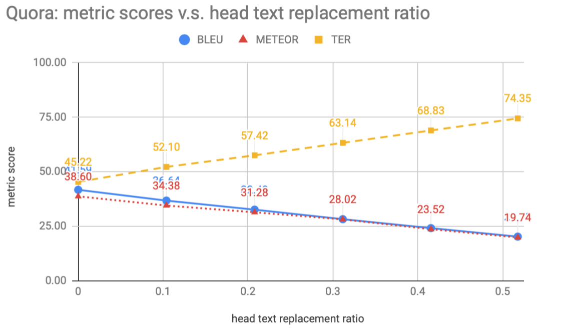

Figure 12: Metric scores v.s. ratio of text that is replaced from the start of the input sentence (Quora)

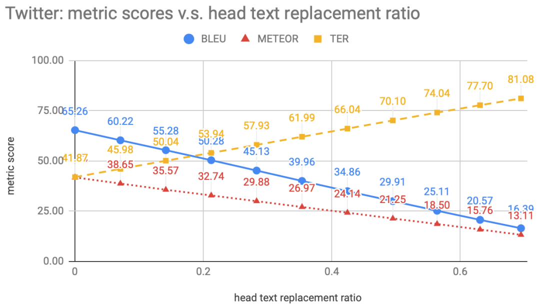

Figure 13: Metric scores v.s. ratio of text that is replaced from the start of the input sentence (Twitter)

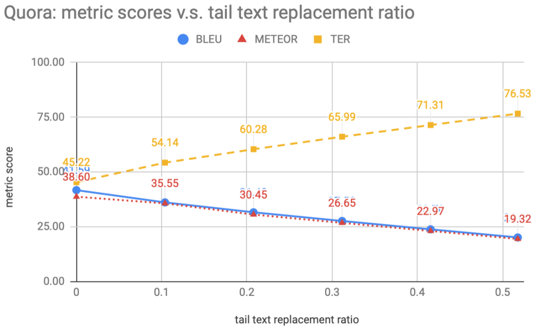

Figure 14: Metric scores v.s. ratio of text that is replaced from the end of the input sentence (Quora)

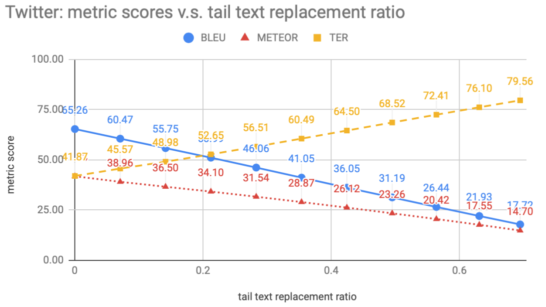

Figure 15: Metric scores v.s. ratio of text that is replaced from the end of the input sentence (Twitter)

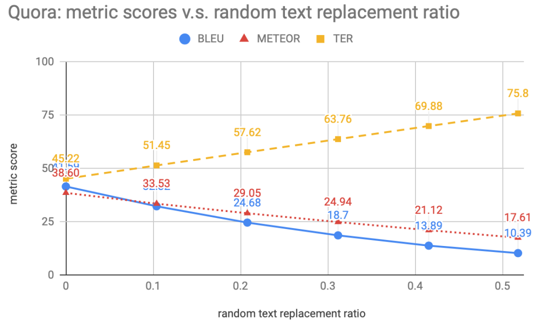

Figure 16: Metric scores v.s. ratio of text that is replaced randomly from the input sentence (Quora)

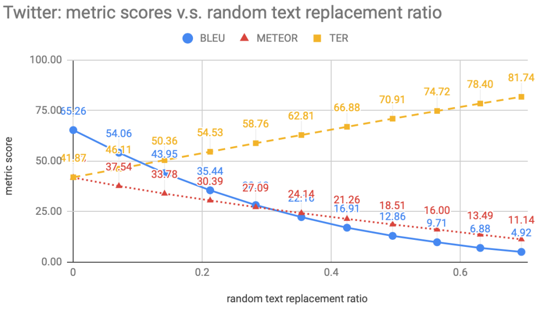

Figure 17: Metric scores v.s. ratio of text that is replaced randomly from the input sentence (Twitter)

C Calculating BLEU score using non-reference sentences

For an input sentence, BLEU scores are usually calculated by comparing the input sentence with a number of reference sentences. We ran experiments in which 5 reference and 100 randomly sampled non-reference sentences were used, and show part of our results below. It can be seen that sentences with higher BLEU scores are more similar to the input sentence, which is to be expected.

Input sentence: what would happen if you hired two private detectives to spy on each other ?

BLEU	
0.3 - 0.35	what will happen if i hire two private detectives to follow each other ? what would happen if i got two private investigators to follow each other ? what would happen if i sent two private investigators to find each other ?
0.25 - 0.3	(None)
0.2 - 0.25	what would happen if earth collided with a black hole ? what if i hired two private eyes and ordered them to follow each other ? what would happen if donald trump lost and refused to concede the election ?
0.15 - 0.2	would i be able to hire two private investigators and then get them to follow each other ? what would happen if donald trump turned out to be a plant for hillary to win the white house ?
0 - 0.15	do i need to pay again on coursera if i switch sessions ? is there anyway to tell if someone blocked you on facebook ? what song do you listen to when you are angry ? ...... what are bugs you noticed on quora ? if i eat a pot cookie , how long until i ’m able to pass a urine test ?

Input sentence: who do you think portrayed batman better : christian bale or ben affleck ?

BLEU	
0.4 - 0.45	according to you , whose batman performance was best : christian bale or ben affleck ?
0.35 - 0.4	(None)
0.3 - 0.35	no fanboys please , but who was the true batman , christian bale or ben affleck ?
0.25 - 0.3	(None)
0.2 - 0.25	(None)
0.15 - 0.2	what do you think about ” chinese dream ” ? what do you think about dota2 ? who was better as batman : bale or affleck ? did ben affleck shine more than christian bale as batman ? do you think that the demonetization policy will backfire for bjp in 2019 elections ?
0 - 0.15	biswapati sarkar : how do you overcome a writer ’s block ? who is the better batman ? affleck or bale ? how do you stop your cat from spraying ? ...... do we always get what we deserve ? can a moon have a moon ?