MVP: Minimal Viable Phrase for Long Text Understanding

Louis Clouâtre[1,2], Amal Zouaq[1], Sarath Chandar[1,2,3]
[1] Polytechnique Montréal
[2] Quebec Artificial Intelligence Institute (Mila)
[3] Canada CIFAR AI Chair
louis.clouatre@mila.quebec, amal.zouaq@polymtl.ca, sarath.chandar@mila.quebec

Abstract

A recent renewal in interest in long text understanding has sparked the emergence of high-quality long text benchmarks, as well as new models demonstrating significant performance improvements on these benchmarks. However, gauging the implication of these advancements based solely on the length of the input text offers limited insight. Such benchmarks may require models to parse long-range dependencies or merely to locate and comprehend the relevant paragraph within a longer text. This work introduces the Minimal Viable Phrase (MVP), a novel metric that determines, through perturbations to the input text, the shortest average text length that needs to be preserved to execute the task with limited performance degradation. Our evaluation of the popular SCROLLS benchmark reveals that only one of its seven tasks necessitates an MVP of over 512 tokens–the maximum text length manageable by the previous generation of pre-trained models. We highlight the limited need for understanding long-range dependencies in resolving these tasks, discuss the specific design decisions that seem to have led to the QuALITY task requiring reliance on long-range dependencies to be solved, and point out specific modeling choices that seem to outperform on the QuALITY task.

Keywords: Explainability, Long-Range Dependencies, Perturbation Studies

1. Introduction

Since the introduction of pretrained Transformers such as BERT (Devlin et al., 2019; Vaswani et al., 2017), the focus of NLP research has predominantly been on short utterances and paragraphs. This focus was partly due to the quadratic complexity of self-attention in relation to sequence length. Renewed interest in research focused on understanding long texts was spurred by recent advancements in machine learning hardware, including improvements in material sciences (Schaller, 1997; Chang et al., 2022) and optimizations of low-level machine learning operations (NVIDIA et al., 2020; Dao et al., 2022), algorithmic improvements to the self-attention mechanism and the growing demand fueled by the widespread adoption of chatbot-style applications like ChatGPT. Supporting this growing interest, several benchmarks have been developed to assess advancements in understanding longer texts (Tay et al., 2020b; Shaham et al., 2022; Hudson and Moubayed, 2022).

To what extent improvements in long text natural language understanding (NLU) benchmarks reflect an enhanced ability to model long-range dependencies is uncertain. Simply considering text length is inadequate to gauge reliance on long text dependencies. For instance, in the scenario of determining a word's definition from a lengthy dictionary, the task might merely involve a straightforward word-matching problem. Without further probing of such benchmarks, the degree to which they depend on extended text sequences remains ambiguous.

Numerous perturbation studies (Pham et al., 2021; Sinha et al., 2021, 2020; Gupta et al., 2021; O'Connor and Andreas, 2021; Clouatre et al., 2022) have been conducted to investigate the importance of specific aspects of text while completing a task. By performing an NLP task after removing the tested aspect from the text, we can understand how essential it was to complete that particular task. For instance, when the order of words in the GLUE benchmark (Wang et al., 2018) is entirely shuffled, the performance impact on most models across various tasks is minimal, suggesting that several GLUE tasks can be addressed using mere bag-of-word information. While one might assume that this benchmark, which measured NLU advancements for years, would require heavy use of the order of words to be completed, verifying such assumptions can yield surprising results.

2. Background and Related Work

This work leverages a set of perturbations designed to remove varying degrees of long-range dependencies from text while retaining other structural elements. We propose the Minimal Viable Phrase (MVP), a novel metric that enables automatic identification of the smallest average contiguous text length essential for task execution without notable performance deterioration. Our study on the widely used SCROLLS benchmark re-

LREC-COLING 2024, pages 12016–12026
20-25 May, 2024.

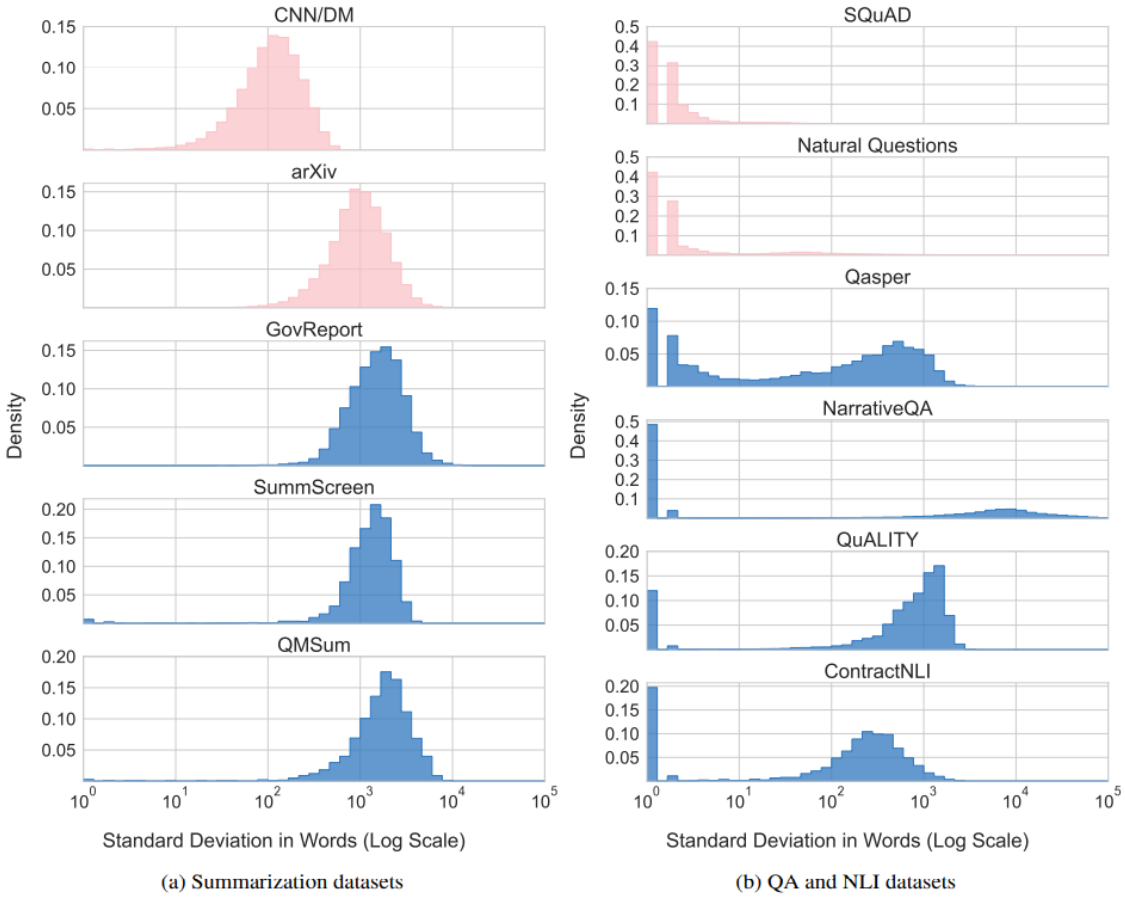

Figure 1: Bi-gram distance in SCROLLS task, as shown in Shaham et al. (2022).

veals that only one task requires an MVP exceeding 512 tokens among the seven tasks included. We discuss the specific choice made in constructing the QuALITY task (Pang et al., 2022) that likely led to it relying on long-range dependencies to be solved, highlight the current limited performance of models in using long-range dependencies, and discuss the likely characteristics that have led certain types of model to outperform other models on the QuALITY task.

2.1. Long Text Benchmarks

Utterance and paragraph length NLP benchmarks, such as the GLUE (Wang et al., 2018) and SuperGLUE benchmark (Sarlin et al., 2020), have proven invaluable resources for evaluating and guiding scientific advancements in NLP. A few long text benchmarks may be suitable for playing a similar role in long-text understanding. As benchmarks become widely accepted, they become defacto targets of scientific advancements. To mitigate Goodhart's law style failing (Strathern, 1997), it is crucial to examine what it is that they measure exactly and to what extent. Our intent is to gauge the extent to which long-text understanding benchmarks currently necessitate an understanding of long-range dependencies. The Long-Range Arena (LRA) benchmark (Tay et al., 2020b), the Multitask Long Document Benchmark (MuLD) (Hudson and Moubayed, 2022), and the Standardized CompaRison Over Long Language Sequences (SCROLLS) Benchmark (Shaham et al., 2022) where all considered for this study.

While widely used, the LRA is mainly composed of non-NLP tasks. The NLP problems are artificially lengthened through byte-level computation, which makes it unclear whether improvements on this benchmark accurately reflect broader improvements in long text NLU.

The MuLD is an NLU benchmark that encompasses six diverse tasks, all involving documents of 10,000 words or more. Some tasks artificially lengthen texts either by expanding the original content or embedding distractors. Most of the tasks in MuLD have an average text length much in excess of what can currently be reasonably handled by pretrained long text models. Paradoxically, this hinders the utilization of this benchmark to identify the extent to which long-range dependencies are relied upon by such models.

The SCROLLS Benchmark is a text-to-text (Raffel et al., 2020) benchmark encompassing seven tasks. It is currently the most extensively employed benchmark for evaluating long text understanding capabilities. The average input lengths for the tasks range from 1706 words to 51,653 words. The benchmark includes a combination of summarization, question-answering (QA), and natural language inference (NLI) tasks.

2.1.1. SCROLLS

SCROLLS provides quantitative analysis suggesting that their sampling of tasks necessitates that a model contends with long-range dependencies that are hundreds to thousands of words in distance on average. To demonstrate this, they take every bi-gram in the reference text and calculate the distance between the first and second part of the bi-gram in the input text, thus showing that if the correct answer were to be present in the text verbatim, it would necessitate fusing far apart portions of the text. Figure 1 shows the analysis result presented in the SCROLLS paper.

We find those metrics and analysis unconvincing in demonstrating that the SCROLLS tasks necessitate longer range dependencies than the other tasks. The arXiv task metric is in line with the summarization tasks present in SCROLLS. The CNN/DailyMail task metric can be explained by the shorter average text length present in this task; the average distance between any random bi-gram would be shorter in absolute terms when measured on a dataset containing shorter sentences on average. The difference between the SCROLLS QA dataset and the SQuAD/Natural Questions dataset is much larger. However, those two datasets are extractive QA problems, meaning that by design, the correct answer is present verbatim in the input text. A comparison with a non-extractive QA dataset would have provided a better reference point for the tasks present in SCROLLS. While this metric does show to what extent the SCROLLS QA tasks are not purely extractive, it is insufficient in showing whether or not long-range dependencies are required to solve those tasks. It is hard to make statements about the extent to which those datasets require an understanding of long-range dependencies to be solved from this metric alone and further exploration is required.

2.2. Long Text Transformers

The Transformer architecture (Vaswani et al., 2017), despite its theoretical potential to attend to infinite sequence lengths, is confined by the quadratic complexity of its self-attention mechanism when it comes to handling longer texts. Various modifications have been proposed to overcome this limitation: introducing sparsity in the attention mechanism (Child et al., 2019a), adopting low-rank approximations for self-attention (Wang et al., 2020a), and a mix of global context with localized attention (Ainslie et al., 2020b; Guo et al., 2021).

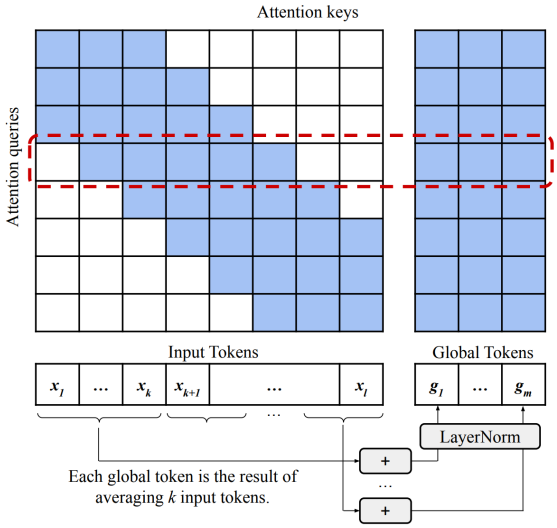

Figure 2: Mix of local and global self-attention, as shown in Guo et al. (2021).

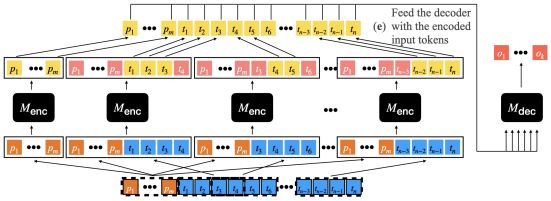

Figure 3: Pooled local self-attention, as shown in Ivgi et al. (2022).

Notably, most successful long-text NLU techniques blend local self-attention with a global context. This context is incorporated either through global tokens (Beltagy et al., 2020; Xiong et al., 2022; Guo et al., 2021)—offering a global view to the localized self-attention—or by employing a decoder, which aggregates the local context generated by the local attention model (Ivgi et al., 2022). An example of both is shown in Figure 2 and Figure 3.

These models can be further separated into two categories:

- **Short-text pooling models**: These models extend existing short-text pretrained models by either aggregating their outputs or adding global tokens to their local attention.

  - The Longformer Encoder-Decoder (LED) (Beltagy et al., 2020), the SLiding Encoder and Decoder (SLED) (Ivgi et al., 2022), and the BART-LS (Xiong et al., 2022) all build on top of a pretrained BART model (Lewis et al., 2019).
  - They build local context with the pretrained BART models, which they either enhance through the use of global tokens (Beltagy et al., 2020; Xiong et al., 2022), which provides global context to the local self-attention mechanism, or with a decoder that can pool the local context built by the BART models (Ivgi et al., 2022).

- **End-to-end models**: These models are pretrained from scratch on long text.
  - LongT5 (Guo et al., 2021; Ainslie et al., 2023) models are pretrained from scratch on long text.
  - They use a mix of local attention and global tokens to give global context to the model while limiting the computational impact of longer sequences.
  - Being pretrained from scratch, they might better understand long-range dependencies than other models since long-range dependencies have been part of their whole pretraining and have only been part of a fraction of the pretraining of the other models, where further pretraining is used.

1.5	The scholar is typesetting. is typeThe schosetting lar.

Figure 4: Subword-level phrase shuffling, has shown in Clouatre et al. (2022).

2.3. Long Range Understanding Studies

Prior work has peered into the limitations of deep learning models' usage of long-range context. Work on the language modeling task with either LSTM's (Khandelwal et al., 2018) or pretrained Transformers (Sun et al., 2021) have studied the effective maximal context length. By limiting how far back the context given to a model was to perform language modeling, they could study the maximal length at which additional context is useful, measured by a drop in test perplexity.

An interesting observation arises when focusing on the prediction of rare words, which intuitively would require more nuanced context from the surrounding text. LSTMs were observed to rely on a context of about 50 tokens for general language modeling but could make use of up to 250 tokens for predicting infrequent words, and pretrained Transformers employed context lengths of up to 2000 tokens for general language modeling, with as many as 5000 tokens being beneficial for predicting rarer words.

Such results are useful in showcasing the importance of using appropriate tasks to evaluate to which extent models can leverage long-range dependencies. Without focusing on the rare word prediction subtask, one might conclude that the effective context that deep learning models can use is 2 to 5 times lower than its effective range. These models may be able to leverage even longer sequences given the right problem, as is showcased in many copying tasks, but without appropriately challenging benchmarks, it is hard to make such conclusions.

2.4. Text Perturbations

Text perturbations to probe the behaviors of trained models is a post-hoc interpretability method (Madsen et al., 2021), providing insights into the inner workings of fully trained models. Such an approach is also inherently linked to the task being perturbed. Any insights obtained from text perturbation approaches are then conditioned on both models and datasets. By removing specific structures of text and evaluating models on these perturbed inputs, it aims to provide insights into what was necessary for the model to complete a particular task.

In our study, we are interested in removing arbitrary amounts of long-range dependencies from the text while limiting the impact of other types of perturbations on the text. We use the Phrase Shuffle (Clouatre et al., 2022), a perturbation of the order of text that aims to preserve as much local structure as possible while removing as much global structure from the text as possible. In other words, phrase shuffling a text will remove an arbitrary amount of long-range dependencies while keeping much of the other aspects of structure in the text intact.

An example of Phrase Shuffling is shown in Figure 4, and the pseudocode to Phrase Shuffling is shown in Figure 1. Phrase Shuffle creates chunks of contiguous tokens of variable length, which are then shuffled. The text is traversed sequentially, from left to right. Every token traversed, with probability ρ, will indicate the end of a phrase, starting a new phrase on the next token. Those phrases are then shuffled, preserving much of the local structure of the text while severing varying degrees of

long-range dependencies.

```
Function PhrasePerturbation(ρ ← 0.5,
 text←list):
    all_phrases ← list();
    phrase ← list(text[0])
    for token in text[1 :] do
        p ∼ Unif([0, 1]);
        if p < ρ then
            all_phrases.append(phrase);
            phrase ← list(token)
        else
            phrase ← [phrase, token];
        end
    end
    all_phrases.append(phrase);
    perturbed_text ←
     ''.join(shuffle(all_phrases))
return perturbed_text
```

Algorithm 1: Pseudocode for Phrase Shuffle. (Clouatre et al., 2022)

3. MVP: Minimal Viable Phrase

The Minimal Viable Phrase (MVP) is defined as the smallest continuous length of text that, on average, needs to be preserved for a model to maintain good performance on a specific task. The process of determining the MVP involves fine-tuning a model on a given task and then evaluating its performance on progressively perturbed inputs through the application of the Phrase Shuffle (Clouatre et al., 2022). Phrase Shuffling builds random contiguous phrases of controllable average length and shuffling those phrases. This preserves much of the structure in the text while severing varying amounts of long-range dependencies.

From this process, we obtain two sets of values: the performances of the model on the different perturbed text as well as the average length of the phrases that were then shuffled. With those values, we apply the Kneedle Algorithm (Satopaa et al., 2011) to determine the MVP of a particular task. The Kneedle Algorithm is a commonly used approach to identifying the "elbows" point in a curve by detecting the point of maximum curvature. The point of maximum curvature detected by the algorithm represents the average phrase length at which the model's performance loss accelerates the most, indicating that the perturbations are starting to affect what is most relied upon by the model to complete the task.

If the models primarily rely on long-range dependencies to complete the task, removing long-range dependencies by shuffling long sub-sequences of the text should have an outsized impact on its performance. However, suppose the point of maximum curvature only happens when phrases become shorter, such as when we break paragraphs into sentences; in that case, we can surmise that a large portion of the performance could be explained not by understanding long-range dependencies but by understanding particular paragraphs or the simple pooling of information gathered from those paragraphs.

4. Experiments

We fine-tune a pretrained Long-T5-Base model, which can handle sequence lengths of up to 16k tokens on every task of the SCROLLS benchmark. We evaluate the fine-tuned model first on the unperturbed test data, then on a series of inputs where we apply varying degrees of phrase shuffling to remove varying degrees of long-range dependencies.

4.1. Training Details

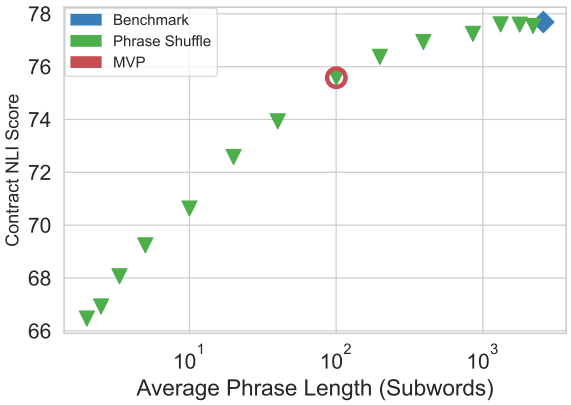

Figure 5: Plotted is the relation between the average phrase length and the performance on the NLI task in the SCROLLS Benchmark. Left is more perturbed; up is better performance. The MVP is circled in red.

In total, we finetuned the model 3 different times on each task. We apply five different random seeds to each perturbation of the text for each trained model. Reported results are than the average over 15 different perturbations obtained from the same parameters. We used the hyperparameters described by Guo et al. (2021) for finetuning. We train on the first 90% of the training set, validate on the last 10%, and test on the validation set as the test labels are not public. We used the TGlobal version of the Long-T5 Base model, meaning the version that uses the Transient global attention. We trained all models for the suggested amount of epochs in the original SCROLLS paper and kept the one with the best validation score.

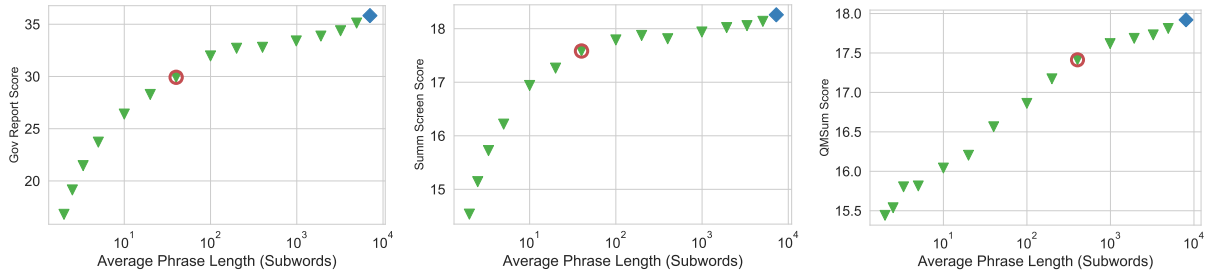

Figure 6: Plotted are the relations between the average phrase length and the performance on the different summarization tasks in the SCROLLS Benchmark. Left is more perturbed; up is better performance. The MVP is circled in red.

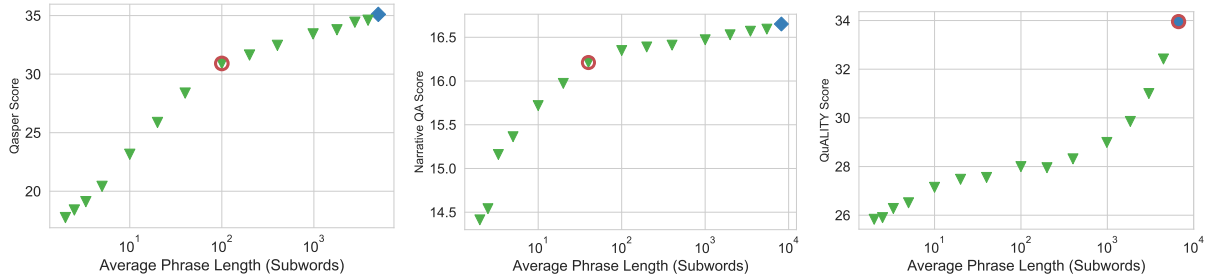

Figure 7: Plotted are the relations between the average phrase length and the performance on the different QA tasks in the SCROLLS Benchmark. Left is more perturbed; up is better performance. The MVP is circled in red.

The ρ values used in the phrase shuffling, meaning the probability that a particular token would be the boundary of a phrase, where: $[0.0001, 0.00025, 0.0005, 0.001, 0.0025, 0.005, 0.01, 0.025, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5]$. Respectively, this would yield average phrase length of lengths: $[10000, 4000, 2000, 1000, 400, 200, 100, 40, 20, 10, 5, 3.33, 2.5, 2.0]$ which covers the whole spectrum of interest. In our experiments, all phrase shuffling is applied to the subwords of the Long-T5 vocabulary and not the words.

5. Results and Discussion

Pictured in Figures 5, 6 and 7 is the general impact of the phrase shuffle on the models' performance, as well as the detected MVPs.

We observe that, QuALITY excluded, none of the tasks in the SCROLLS benchmark have an MVP above 512 tokens, a length of text that does not require a long text model to handle. In all non-QuALITY tasks, we observe very little impact on performance from breaking down the text into what is, in effect, paragraphs and sentences. This may explain the popularity and success of short-text pooling models for long text understanding, as we see limited impact from removing longer-range dependencies on most evaluated tasks.

The results and tasks are summarized in Table 1. From those results, we cannot find an obvious relation between the text length, the task type, and the MVP of a task.

Task	Average Length (Word)	Task Type	MVP (Tokens)
GovReport	7886	Summarization	40
SummScreenFD	5598	Summarization	20
QMSum	9497	Summarization	400
Qasper	3629	QA	100
NarrativeQA	51653	QA	40
QuALITY	4193	QA	**6671**
ContractNLI	1706	NLI	100

Table 1: Summary information of the different tasks used and their MVPs.

5.1. Model Comparison

From the SCROLLS public leaderboard, we can observe that approaches such as SLED and BART-LS that rely on pretrained short-context building blocks will, relative to their scores on the other tasks, systematically under-perform models that are trained from scratch on long text, such as Long-T5, on QuALITY. In Table 2, we compare the results of BART-LS, BART-Large SLED, and LongT5 Base. Those three models were chosen for comparison as they have fairly analogous parameter counts as well as average performance. While the aggregate score of the non-QuALITY task on both BART-LS and BART-Large SLED are

either on par or above LongT5 Base, they score lower by a margin on the QuALITY task. This lends credence to the hypothesis that long-range dependencies are especially important in the QuALITY task and less so in the other tasks.

Of the tasks on which we have human performance, QuALITY is the one in which neural models are the furthest from human level. We summarize the information on human performance on the SCROLLS benchmark in Table 3. This neural model to human gap is likely caused by the limited ability of those models to properly understand long-range dependencies, which would be most necessary in the QuALITY task.

Model	Non QuALITY avg	QuALITY
BART-LS	40.4	35.9
BART-Large SLED	38.52	34.8
LongT5 Base	38.83	**37.25**

Table 2: Comparable models scores on QuALITY and non-quality tasks. LongT5 Base performs on part with BART-Large SLED and underperforms BART-LS on the aggregate of the non-QuALITY tasks but outperforms them by a margin on the QuALITY task.

Task	Human Performance	Best Neural Approach	Human Neural Gap
Qasper	60.9	53.9	7
QuALITY	93.5	48.1	45.4
NarrativeQA	58.7	31.1	27.6

Table 3: Estimated human performance where available, with best public results for comparison. (Shaham et al., 2022)

5.2. QUaLITY

In the construction of their dataset, QUaLITY introduced the speed validation. The overall process is pictured in Figure 8. Annotators are given 45 seconds to read the context paragraph and answer the question quickly. Suppose a human cannot answer a question under a certain time threshold that can readily be answered given an unlimited amount of time. In that case, we can ensure that more than simply skimming is needed and that any single passage through the text is unlikely to permit us to complete the task. Writers who built the dataset were incentivized to produce questions that annotators in speed validation would get wrong but annotators with unlimited time would get right. Half of the QUaLITY dataset is made up of such questions. We believe that this step is likely the main differentiator between QuALITY and the rest of the tasks and can generally be adapted to many other task constructions.

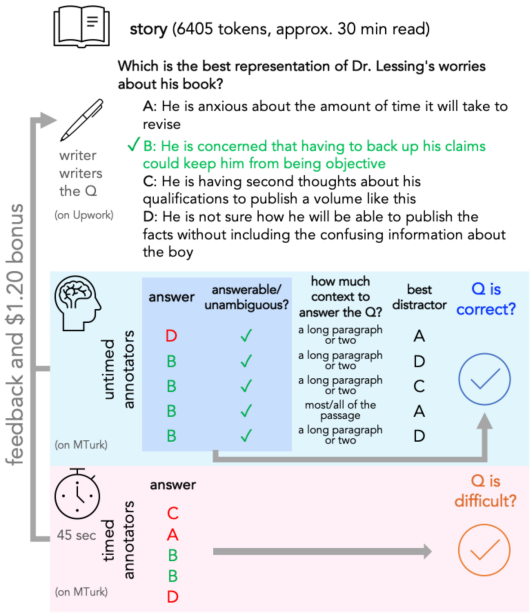

Figure 8: Crowdsourcing pipeline used to build the QUaLITY dataset (Pang et al., 2022).

Pre-existing tasks could be filtered down with this approach. One could take existing test sets of tasks such as NarrativeQA or Qasper, apply the speed validation described in the QUaLITY paper wholesale, and provide a subset of those tasks that would both be validated to not be trivial for human to complete as well as being more likely to require leveraging long-range dependencies to complete. This would provide a lot of clarity as to which extent our current benchmarks rely on long-range dependencies.

6. Limitation

There are several limitations to this study that should be noted.

First, while the method used is effective for distinguishing between tasks that require strict long-range dependencies of a certain length and those that do not, it is less effective at distinguishing between tasks that require the synthesis of smaller units, such as paragraphs, and tasks where a single small unit of text suffices. This is an important distinction since tasks that can be described as search problems, such as finding a word's definition in a dictionary, may be better served by small-context models, while synthesis tasks, like summarization, will still benefit from models that can pool information from the larger context.

Second, due to hardware and model limitations, we were restricted to exploring a single model and using text of sequence lengths of up to 16,000 to-

kens, which limits the scope of our empirical results. Larger models and benchmarks such as MULD, which contains much longer texts on average, should be explored to add to our conclusions.

Third, the use of the Kneedle Algorithm to find the inflection point is not without issues. We use it to find the point of maximum curvature between the score of our model and the logarithm of the phrase length. While using a logarithmic scale seems intuitively sound — for instance, the impact difference between average phrase lengths of 10,000 and 5,000 tokens should parallel that between 100 and 50 tokens — it is ultimately arbitrary, and the Kneedle Algorithm is sensitive to such decisions. The use of the elbow of a curve itself can have issues. While widely used in several domains to find points of interest, it remains fairly arbitrary, relying mostly on intuition for justification rather than theory.

7. Conclusion

In this work, we have introduced the Minimal Viable Phrase, which gives us information on how long-range dependencies are relied upon by a model to complete a task. We have found that only the QuALITY task relies strictly on long-range dependencies to be completed. We speculate that the specific design choice made in building the task, specifically the time trial and the writers' incentives, ensured that long-range dependencies were central to the task completion. From our results, we believe that while constructing benchmarks for long text understanding, special attention to how the different tasks are constructed concerning long-range dependencies is warranted. It does not seem that long-text, even if from tasks that should intuitively require understanding some long-range dependencies such as summarization, is a sufficient criterion to ensure that long-range dependencies understanding is properly evaluated. We may not notice serious shortcomings in our approaches if the scientific community optimizes for benchmarks that include insufficient long-range dependencies.

8. Bibliographical References

Alfred V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling*, volume 1. Prentice-Hall, Englewood Cliffs, NJ.

Joshua Ainslie, Tao Lei, Michiel de Jong, Santiago Ontan'on, Siddhartha Brahma, Yury Zemlyanskiy, David C. Uthus, Mandy Guo, James Lee-Thorp, Yi Tay, Yun-Hsuan Sung, and Sumit K. Sanghai. 2023. Colt5: Faster long-range transformers with conditional computation. *ArXiv*, abs/2303.09752.

Joshua Ainslie, Santiago Ontañón, Chris Alberti, Philip Pham, Anirudh Ravula, and Sumit Sanghai. 2020a. ETC: encoding long and structured data in transformers. *CoRR*, abs/2004.08483.

Joshua Ainslie, Santiago Ontañón, Chris Alberti, Philip Pham, Anirudh Ravula, and Sumit K. Sanghai. 2020b. Etc: Encoding long and structured data in transformers. *ArXiv*, abs/2004.08483.

American Psychological Association. 1983. *Publications Manual*. American Psychological Association, Washington, DC.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Galen Andrew and Jianfeng Gao. 2007. Scalable training of L_1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, pages 33–40.

Anthony McEnery and others. 2004. *The EMILLE/CIIL Corpus*. EMILLE (Enabling Minority Language Engineering) Project. distributed via ELRA: ELRA-Id W0037, ISLRN 039-846-040-604-0.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

BSI. 1973a. *Natural Fibre Twines*, 3rd edition. British Standards Institution, London. BS 2570.

BSI. 1973b. Natural fibre twines. BS 2570, British Standards Institution, London. 3rd. edn.

A. Castor and L. E. Pollux. 1992. The use of user modelling to guide inference and learning. *Applied Intelligence*, 2(1):37–53.

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the Association for Computing Machinery*, 28(1):114–133.

Chih-Hao Chang, V.S. Chang, K.H. Pan, K.T. Lai, J. H. Lu, J.A. Ng, C.Y. Chen, B.F. Wu, C.J. Lin, C.S. Liang, C.P. Tsao, Y.S. Mor, C.T. Li, T.C. Lin, C.H. Hsieh, P.N. Chen, H.H. Hsu, J.H. Chen, H.F. Chen, J.Y. Yeh, M.C. Chiang, C.Y. Lin, J.J. Liaw, C.H. Wang, S.B. Lee, C.C. Chen, H.J. Lin, R. Chen, K.W. Chen, C.O. Chui, Y.C. Yeo, K.B. Huang, T.L. Lee, M.H. Tsai, K.S. Chen, Y.C. Lu, S.M. Jang, and S.-Y. Wu. 2022. Critical

process features enabling aggressive contacted gate pitch scaling for 3nm cmos technology and beyond. In *2022 International Electron Devices Meeting (IEDM)*, pages 27.1.1–27.1.4.

J.L. Chercheur. 1994. *Case-Based Reasoning*, 2nd edition. Morgan Kaufman Publishers, San Mateo, CA.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019a. Generating long sequences with sparse transformers. *ArXiv*, abs/1904.10509.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019b. Generating long sequences with sparse transformers.

N. Chomsky. 1973. Conditions on transformations. In *A festschrift for Morris Halle*, New York. Holt, Rinehart & Winston.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. 2022. Rethinking attention with performers.

Louis Clouatre, Prasanna Parthasarathi, Amal Zouaq, and Sarath Chandar. 2022. Local structure matters most: Perturbation study in nlu. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3712–3731.

James W. Cooley and John W. Tukey. 1965. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Umberto Eco. 1990. *The Limits of Interpretation*. Indian University Press.

Mandy Guo, Joshua Ainslie, David C. Uthus, Santiago Ontañón, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2021. Longt5: Efficient text-to-text transformer for long sequences. *CoRR*, abs/2112.07916.

Ashim Gupta, Giorgi Kvernadze, and Vivek Srikumar. 2021. Bert & family eat word salad: Experiments with text understanding. *arXiv preprint arXiv:2101.03453*.

Dan Gusfield. 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, Cambridge, UK.

Paul Gerhard Hoel. 1971a. *Elementary Statistics*, 3rd edition. Wiley series in probability and mathematical statistics. Wiley, New York, Chichester. ISBN 0 471 40300.

Paul Gerhard Hoel. 1971b. *Elementary Statistics*, 3rd edition, Wiley series in probability and mathematical statistics, pages 19–33. Wiley, New York, Chichester. ISBN 0 471 40300.

G Thomas Hudson and Noura Al Moubayed. 2022. Muld: The multitask long document benchmark.

Maor Ivgi, Uri Shaham, and Jonathan Berant. 2022. Efficient long-text understanding with short-text models.

Otto Jespersen. 1922. *Language: Its Nature, Development, and Origin*. Allen and Unwin.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention.

Khalid Choukri and Niklas Paullson. 2004. *The OrienTel Moroccan MCA (Modern Colloquial Arabic) database*. distributed via ELRA: ELRA-Id ELRA-S0183, ISLRN 613-578-868-832-2.

Urvashi Khandelwal, He He, Peng Qi, and Dan Jurafsky. 2018. Sharp nearby, fuzzy far away: How neural language models use context. In *Annual Meeting of the Association for Computational Linguistics*.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer.

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.

Andreas Madsen, Siva Reddy, and Sarath Chandar. 2021. Post-hoc interpretability for neural NLP: A survey. *CoRR*, abs/2108.04840.

NVIDIA, Péter Vingelmann, and Frank H.P. Fitzek. 2020. Cuda, release: 10.2.89.

Joe O'Connor and Jacob Andreas. 2021. What context features can transformer language models use? In *ACL/IJCNLP*.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, and Samuel R. Bowman. 2022. Quality: Question answering with long input texts, yes!

Thang M. Pham, Trung Bui, Long Mai, and Anh M Nguyen. 2021. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? *ArXiv*, abs/2012.15180.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

Mohammad Sadegh Rasooli and Joel R. Tetreault. 2015. Yara parser: A fast and accurate dependency parser. *Computing Research Repository*, arXiv:1503.06733. Version 2.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. Superglue: Learning feature matching with graph neural networks.

Ville Satopaa, Jeannie R. Albrecht, David E. Irwin, and Barath Raghavan. 2011. Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In *ICDCS Workshops*, pages 166–171. IEEE Computer Society.

R.R. Schaller. 1997. Moore's law: past, present and future. *IEEE Spectrum*, 34(6):52–59.

Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, and Omer Levy. 2022. SCROLLS: standardized comparison over long language sequences. *CoRR*, abs/2201.03533.

Charles Joseph Singer, E. J. Holmyard, and A. R. Hall, editors. 1954–58. *A history of technology*. Oxford University Press, London. 5 vol.

Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. 2021. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. *arXiv preprint arXiv:2104.06644*.

Koustuv Sinha, Prasanna Parthasarathi, Joelle Pineau, and Adina Williams. 2020. Unnatural language inference. *arXiv preprint arXiv:2101.00010*.

Speecon Consortium. 2011. *Catalan Speecon database*. SpeeCon. Speecon Project, distributed via ELRA: ELRA-Id S0327, Speecon resources, 1.0, ISLRN 935-211-147-357-5.

Marilyn Strathern. 1997. 'improving ratings': audit in the british university system. *European Review*, 5(3):305–321.

Jannik Strötgen and Michael Gertz. 2012. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3746–3753, Istanbul, Turkey. European Language Resource Association (ELRA).

Simeng Sun, Kalpesh Krishna, Andrew Mattarella-Micke, and Mohit Iyyer. 2021. Do long-range language models actually use long-range context? *ArXiv*, abs/2109.09115.

S. Superman, B. Batman, C. Catwoman, and S. Spiderman. 2000. *Superheroes experiences with books*, 20th edition. The Phantom Editors Associates, Gotham City.

Yi Tay, Dara Bahri, Donald Metzler, Da-Cheng Juan, Zhe Zhao, and Che Zheng. 2021. Synthesizer: Rethinking self-attention in transformer models.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. 2020a. Sparse sinkhorn attention. *CoRR*, abs/2002.11296.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2020b. Long range arena: A benchmark for efficient transformers. *CoRR*, abs/2011.04006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. 2020a. Linformer: Self-attention with linear complexity. *ArXiv*, abs/2006.04768.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. 2020b. Linformer: Self-attention with linear complexity. *CoRR*, abs/2006.04768.

Wenhan Xiong, Anchit Gupta, Shubham Toshniwal, Yashar Mehdad, and Wen tau Yih. 2022. Adapting pretrained text-to-text models for long text sequences.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2021. Big bird: Transformers for longer sequences.

Kai Zhang, Chongyang Tao, Tao Shen, Can Xu, Xiubo Geng, Binxing Jiao, and Daxin Jiang. 2023. Led: Lexicon-enlightened dense retriever for large-scale retrieval.