TextGenSHAP: Scalable Post-hoc Explanations in Text Generation with Long Documents

James Enouen[1,*], **Hootan Nakhost**[2], **Sayna Ebrahimi**[2], **Sercan Ö Arik**[2], **Yan Liu**[1], **Tomas Pfister**[2]
[1]University of Southern California, Los Angeles, CA
[2]Google Cloud AI Research, Sunnyvale, CA

Abstract

Large language models (LLMs) have attracted great interest in many real-world applications; however, their "black-box" nature necessitates scalable and faithful explanations. Shapley values have matured as an explainability method for deep learning, but extending them to LLMs is difficult due to long input contexts and autoregressive output generation. We introduce TextGenSHAP, an efficient post-hoc explanation method incorporating LLM-specific techniques, which leads to significant runtime improvements: token-level explanations in minutes not hours, and document-level explanations within seconds. We demonstrate how such explanations can improve end-to-end performance of retrieval augmented generation by localizing important words within long documents and reranking passages collected by retrieval systems. On various open-domain question answering benchmarks, we show TextGenSHAP improves the retrieval recall and prediction accuracy significantly.

1 Introduction

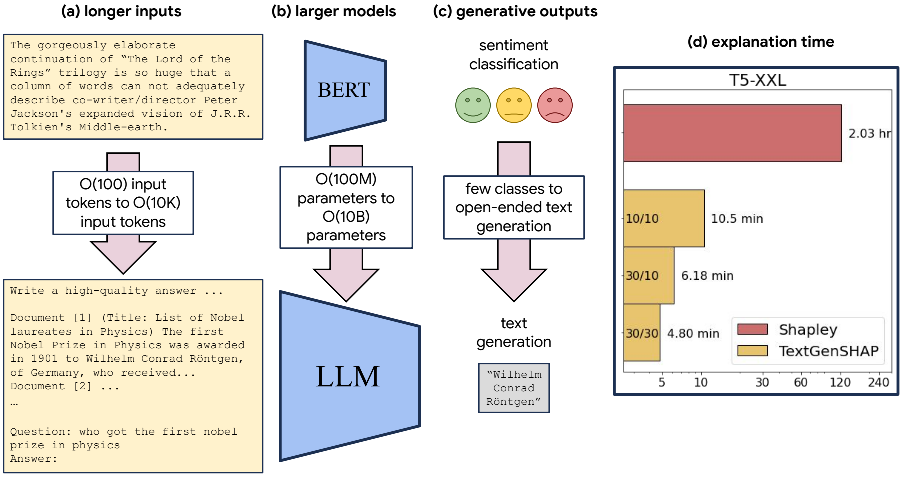
Large language models (LLMs) continue to rapidly excel at different text-generation tasks alongside the continued growth of resources dedicated to training text-based models (Brown et al., 2020; Chowdhery et al., 2022; Touvron et al., 2023). LLM's impressive capabilities have led to their widespread adoption throughout academic and commercial applications. Their capacity to reason cohesively on a wide range of natural language processing (NLP) tasks has motivated further efforts to enable a single model to automatically ingest increasingly large contexts. These long-context models have been shown to improve zero-shot, few-shot, and retrieval-augmented performance via in-context learning (Izacard et al., 2022b; Huang et al., 2023a; Ram et al., 2023) and to reduce the need for training task-specific models, empowering non-experts to readily use LLMs.

Despite their remarkable text generation capabilities, LLMs which are trained to model statistical correlations in language can offer only limited insight into their internal mechanisms. Accordingly, LLMs are widely considered black-box models which are incredibly difficult to explain and understand. In the wake of widespread adoption amongst the general population, challenges beyond LLM prediction performance including safety, security, and truthfulness have gained increasing prominence. Growing reports of hallucinated material, harmful counseling, prejudiced content, and other real-world consequences continue to raise concerns. Often, *explainability* is hailed as a crucial avenue for addressing these concerns, enabling insights into the model's decision-making process and allowing stakeholders to directly scrutinize the reasoning behind unsafe or untruthful responses.

Recent surveys in explainability for NLP juxtapose the two main criteria for model explanations: understandability and faithfulness (Lyu et al., 2023a; Zhao et al., 2023; Mosca et al., 2022). Understandability refers to how easily an explanation is understood by a human user, whereas faithfulness measures how accurately it reflects the model's reasoning process. Effectively balancing these objectives for a given explanation technique remains an ongoing challenge (Rudin, 2019). Popular explanation approaches like attention scores, gradient saliency, and self-explained reasoning are generally considered to be understandable; however, ongoing debates question whether these approaches also provide high-fidelity explanations (Jain and Wallace, 2019; Adebayo et al., 2018; Ghorbani et al., 2019; Wang et al., 2020; Wei et al., 2022). For tabular and image data, the Shapley value (Lundberg and Lee, 2017) stands out due to its strong theoretical foundations, grounded in

[*] enouen@usc.edu

Findings of the Association for Computational Linguistics ACL 2024, pages 13984–14011
August 11-16, 2024 ©2024 Association for Computational Linguistics



The gorgeously elaborate continuation of "The Lord of the Rings" trilogy is so huge that a column of words can not adequately describe co-writer/director Peter Jackson's expanded vision of J.R.R. Tolkien's Middle-earth.

```
Write a high-quality answer ...

Document [1] (Title: List of Nobel
laureates in Physics) The first
Nobel Prize in Physics was awarded
in 1901 to Wilhelm Conrad Röntgen,
of Germany, who received...
Document [2] ...
…

Question: who got the first nobel
prize in physics
Answer:
```

Figure 1: Post-hoc explainability generation gets more challenging for: (a) longer inputs, (b) larger models, and (c) open-ended text generation. These lead to significantly increased times for extracting explanations (d) which can be prohibitively long for human-in-the-loop model improvement.

axioms guaranteeing: symmetry, efficiency, nullity, and linearity (Mitchell et al., 2022). In the NLP domain, however, faithful approaches like the Shapley value suffer greatly in their ability to scale to larger models and longer inputs, leading to impractically long wait times for explanations in LLM workflows.

To address the limitations of current explainability methods in the realm of NLP, we introduce TextGenSHAP, a novel approach to extend Shapley values for text generation while keeping a computational speed more suitable for real-world LLM use cases. We focus on explanations in the challenging scenario of open-ended text generation using long inputs as prompts, specifically focusing on the task of abstractive question answering from retrieval-augmented documents. We leverage well-founded historical works in game theory to support our new Shapley score's definition, and then develop an efficient algorithm designed specifically for transformer models. Accordingly, we demonstrate our method's scalability to new applications across three key aspects shown in Fig. 1: (a) handling longer contexts with thousands of input tokens using the hierarchical structure of natural text; (b) accommodating larger models with billions of parameters using hardware-aware speedups; and (c) explaining free-form text generation, instead of only discriminative tasks like classification (which were the focus of previous work.)

Furthermore, we demonstrate how the explanations generated by our TextGenSHAP can enhance the performance of open-domain question answering on both MIRACL and NQ-Open, enhancing the recall of document retrieval systems by multiple points and closing the accuracy gap of open-domain question answering with a 5-10% point improvement.

2 Related Work

Post-hoc Model Explainability. There have been many works focusing on explanations which show how machine learning models utilize their input features to make predictions. Notable post-hoc explanation approaches include LIME (Ribeiro et al., 2016), SHAP (Lundberg and Lee, 2017), and Integrated Gradients (Sundararajan et al., 2017), although SHAP and Shapley have recently become dominant due to their strong foundations. For NLP, many related perturbation-based methods also exist, leveraging the hierarchical structure and sequential order of text (Chen et al., 2019; Jin et al., 2020; Chen et al., 2020). More recent methods extend beyond binary classification tasks with contrastive versions of the original techniques (Jacovi et al., 2021; Yin and Neubig, 2022). However, none of these existing works tackle non-binary hierarchies or generative text, which we identify as key challenges overcome by our approach, see Sec. 3.3 and 3.2 respectively. Although existing work has looked at accelerating Shapley value estimation (Jethani et al., 2022) for tabular and image data types, it remains challenging to extend such approaches to NLP because of generative text outputs. Specifically, all existing methods require prespecification of candidate outputs and cannot be applied to the large output spaces of free-form text generation. Accordingly, current post-hoc methods for text generation are limited to the sequential application of tabular approaches (Sarti et al., 2023).

Self-explanations and Rationales. For NLP explanations, another popular approach is training models generating 'rationales' to highlight important tokens for prediction, often by aligning with rationales collected either from human annotators (Arous et al., 2021; Joshi et al., 2022) or post-hoc explanations (Stacey et al., 2022; Chan et al., 2022). Still, such approaches remain mostly limited to classification tasks instead of generative tasks, likely due to both the difficulties in collecting human rationales and the nonexistence of post-hoc explanations discussed above.

Natural language explanations, such as chain-of-thought (Wei et al., 2022), where LLMs emit explanations about themselves are hence some of the only available explanations for text generation. Unfortunately, such prompt-engineering and scratchpad approaches are only part of the mechanistic process of generation and provide no guarantees on faithfulness or explanation accuracy. Ongoing work aims to measure the degree of faithfulness which could be provided by such explanations (Jacovi and Goldberg, 2021; Zheng et al., 2022; Lyu et al., 2023b; Lanham et al., 2023).

Information Retrieval from Long Documents. Question answering (QA) is a fundamental NLP task, evolving from reading comprehension into retrieval-augmented fusion with increasingly large knowledge bases. As early as the NQ dataset (Kwiatkowski et al., 2019), the bifurcation between the original long-document format (entire Wikipedia page) and the open-domain format (all of Wikipedia) had already emerged (Lee et al., 2019; Karpukhin et al., 2020). [1] Open-domain QA is dominated by pipelined approaches where fast retrievers rank relevant passages for slower, more thorough reader models. Recently, neural-based retrievers have emerged for this first stage, uprooting the long reign of term-frequency approaches (Izacard et al., 2022a; Karpukhin et al., 2020; Ma et al., 2021; Formal et al., 2021; Guu et al., 2020; Mao et al., 2021; Johnson et al., 2019). Simultaneously, improvements have been made on the reader model side of the pipelined approach with Fusion-in-Decoder (FiD) (Izacard and Grave, 2021b,a) designing an efficient QA architecture and 'Lost in the Middle' (LitM) (Liu et al., 2023) identifying the reader's brittleness to passage order.

Architectures for long inputs. In pursuit of the impressive capabilities of large-scale, end-to-end training, there has also been a surge in architectures which can increase the context size of LMs. Maximum context windows have quickly expanded from thousands of tokens to many millions of tokens with the use of efficient sparsity methods (Wu et al., 2022; Bulatov et al., 2022; Ding et al., 2023). Two main approaches exist: methods utilizing sparsity which closely mimicks that of information-retrieval for relevant tokens or with external memory (Bertsch et al., 2023; Wu et al., 2022; Bulatov et al., 2022, 2023; Johnson et al., 2019), and methods instead using block sparse attention matrices to reduce the necessary computations of the attention mechanism (Beltagy et al., 2020; Zhang et al., 2022a; Ding et al., 2023; Dao et al., 2022).

3 Explainability Framework

Notation. Consider an LLM using a vocabulary of size $V \in \mathbb{N}$ for input sequences $\boldsymbol{x} \in \mathcal{X} := [V]^d$ and output sequences $\boldsymbol{y} \in \mathcal{Y} := [V]^m$ with input length $d \in \mathbb{N}$ and maximum output length $m \in \mathbb{N}$, where $[V] := \{1, \ldots, V\}$. Broadly, a text-generation model takes an input sequence of tokens and defines a probability vector over all possible outputs, $F : \mathcal{X} \to [0,1]^{\mathcal{Y}}$. Hence, we have $F(\boldsymbol{x})_{\boldsymbol{y}}$ denote $\boldsymbol{y}$'s probability of being generated given $\boldsymbol{x}$.

To enable explanation via feature attribution methods like the Shapley value, we need to be able to mask certain subsets of the input tokens. Let $\boldsymbol{s} \in \mathcal{M} := \{0,1\}^d$ be a binary mask on the input tokens. We next define a masked text-generation model, $f : \mathcal{X} \times \mathcal{M} \to [0,1]^{\mathcal{Y}}$, which takes both an input sequence and an input mask. In practice, we replace all input tokens which are not in the mask $\boldsymbol{s}$ by the <pad> token before inputting it to the model. If we assume the <pad> or <mask> token is taken to be $p \in [V]$ and identify the d-vector composed of all p to be $\boldsymbol{p}$, then we can write this as $f(x, \boldsymbol{s}) := F(\boldsymbol{x} \odot \boldsymbol{s} + \boldsymbol{p} \odot (1 - \boldsymbol{s}))$.

3.1 Shapley Value

Shapley values, originally derived to allocate the worth of individual players in a cooperative game, have since become a dominant paradigm for explaining feature attributions of black-box models (Shapley, 1953; Lundberg and Lee, 2017). In Sec.

[1] Unfortunately, this bifurcation leads to conflicting nomenclature for 'document'. In the former, it is the entire long document. In the latter, it is the unit of retrieval.

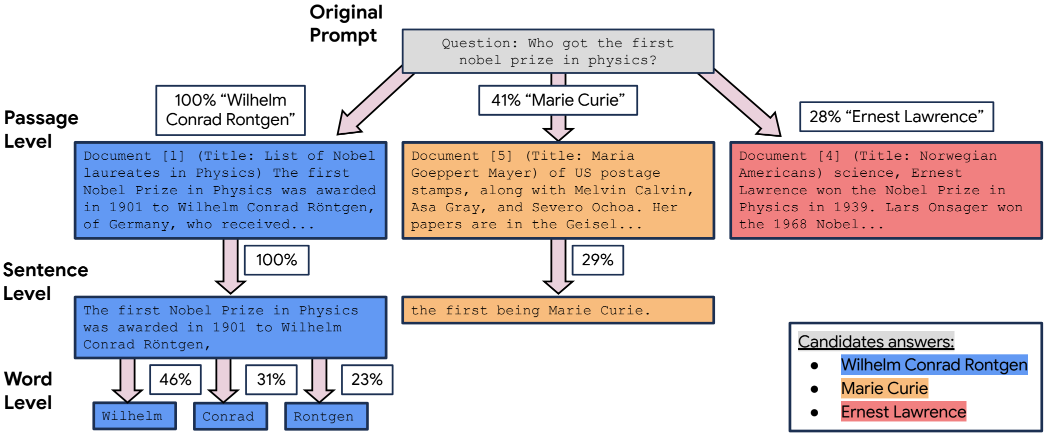

Original Prompt
Question: Who got the first nobel prize in physics?
100% "Wilhelm Conrad Rontgen"
41% "Marie Curie"
28% "Ernest Lawrence"
Passage Level
Document [1] (Title: List of Nobel laureates in Physics) The first Nobel Prize in Physics was awarded in 1901 to Wilhelm Conrad Röntgen, of Germany, who received...
Document [5] (Title: Maria Goeppert Mayer) of US postage stamps, along with Melvin Calvin, Asa Gray, and Severo Ochoa. Her papers are in the Geisel...
Document [4] (Title: Norwegian Americans) science, Ernest Lawrence won the Nobel Prize in Physics in 1939. Lars Onsager won the 1968 Nobel...
100%
29%
Sentence Level
The first Nobel Prize in Physics was awarded in 1901 to Wilhelm Conrad Röntgen,
the first being Marie Curie.
Candidates answers:
Wilhelm Conrad Rontgen
Marie Curie
Ernest Lawrence
Word Level
46%
31%
23%
Wilhelm
Conrad
Rontgen

Figure 2: Graphic portraying the hierarchical explanations generated by TextGenSHAP. Different colors correspond to output sequences generated by the model. Percentages correspond to Shapley-Shubik probabilities of a passage/sentence/word to influence the model's decision under the Shapley distribution. It is observed that the model is more likely to choose other Nobel Prize winners in the absence of the true winner. Exploration is stopped if not reaching the 30% threshold.

3.2, we describe the Shapley-Shubik and Penrose-Banzhaf values which are extended to work for voting games (Shapley and Shubik, 1954; Banzhaf, 1965; Penrose, 1946). In Sec. 3.3, we describe the hierarchical extension, the Owen-Winter value (Owen, 1977; Winter, 2002). We use these two extensions to help overcome the challenges posed in Fig. 1c and 1a, respectively.

To define the 'value functions' required to define the Shapley score in a way that is consistent with the existing interpretability literature, we must first correspond our binary input masks with input-feature subsets. In particular, for any element of the power set $S \in \mathcal{P}([d]) := \{S \subseteq [d]\}$, there is a unique corresponding binary mask $\boldsymbol{s} \in \{0,1\}^d$ via the indicator function $\boldsymbol{s} = 1_S$. For any input token $i \in [d]$, we will use the set notation $(S+i) := S \cup \{i\}$ and $(S-i) := S \backslash \{i\}$ to unmask or mask the token. For a fixed x, we will then write $v_\ell(S) := \log(f(x, 1_S))$ and $v_p(S) := f(x, 1_S)$ as our two candidate value functions (log-likelihood and likelihood).

The Shapley value is formulated as an expectation over uniformly distributed permutations:

$$\varphi_i = \mathbb{E}_\pi \left[v_\ell(S_{\pi,i} + i) - v_\ell(S_{\pi,i} - i) \right], \qquad (1)$$

where $\pi : [d] \to [d]$ denotes the sampled permutation, representing a random order of the features (tokens) and $S_{\pi,i} := \{j \in [d] : \pi(j) < \pi(i)\}$ is the set of elements which precede i in the order defined by π. Hence, $S_{\pi,i} + i = \{j \in [d] : \pi(j) \leq \pi(i)\}$ and $S_{\pi,i} - i = S_{\pi,i} = \{j \in [d] : \pi(j) < \pi(i)\}$, where we unnecessarily subtract the element i in preparation for Section 3.2. We follow the standard approach of permutation sampling to estimate the Shapley value as the empirical mean over a finite set of sampled permutations (Covert et al., 2021).

The key challenge of applying the conventional Shapley formulation is that we do not have access to the full probability vector $F(x)$, which is of exponentially large size. For previous work in classification tasks, the log-probabilities may be computed exactly for every candidate output. In open-ended text generation, however, we utilize sequential decoding algorithms like greedy decoding and K-beam generation to recover only a sparse subset of the exponentially large probability vector $F(x) \in [0,1]^{[V]^m}$. In the next section, we show how to adapt Shapley to handle generated text coming from distributions of a-priori unknown support.

3.2 Extension to Generative Outputs

Although the Shapley value has found wide success in tasks like classification and regression, it struggles to be applied to generative tasks using sequential decoding. Towards this end, we leverage the voting theory reformulation of the conventional Shapley value, called the Shapley-Shubik power index. We consider each input token as a 'voter' casting a vote for a generated answer, aiming to 'elect' their preferred answer under the LM's black-box voting system. While conventional Shapley employs a value function represented as the vector of log-probabilities, Shapley-Shubik formulation operates on the probability vector. Hereafter, we will refer to the 'Shapley-Shubik power index' as

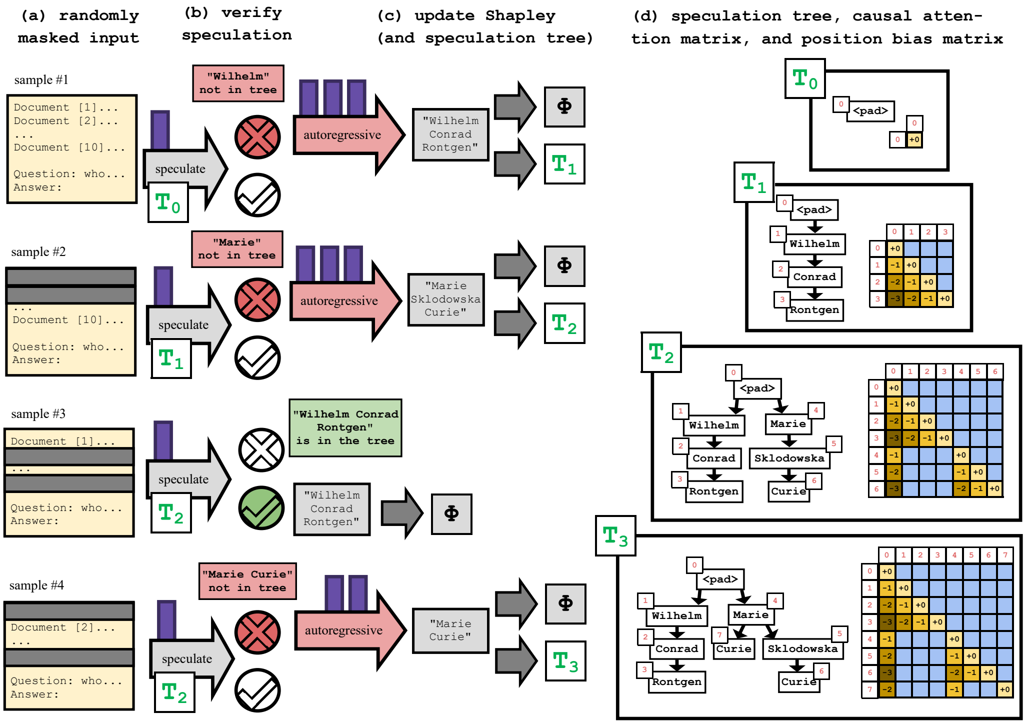


Figure 3: Visualization of how to use the speculative decoding approach proposed in TextGenSHAP to improve the resampling algorithm speed. (a) The randomly masked inputs generated to calculate the Shapley value. (b) Running the decoder a single time with the speculation tree and then verifying whether the true output is within the speculated output. (c) If the speculation is rejected, we must run the decoder autoregressively to generate the correct output. Each purple bar represents a single time we call the decoder. Afterwards we update the Shapley value and add the new output to the speculation tree. If the speculation is accepted, we update the Shapley value with the correctly speculated output. (d) As we run the algorithm, we keep track of the speculation tree and its position bias matrix. The causal attention mask can be computed directly from the position bias matrix by masking out all blue entries and only keeping yellow entries. The causal attention matrix quickly takes a more complex form than the typical triangular matrix to correctly compute the output likelihoods.

still just 'Shapley' for brevity. We can equivalently reformulate Shapley as an expectation over a random subset instead of over a random permutation, highlighting its connection with the Banzhaf value:

$$\varphi_i^{Sh} := \mathbb{E}_{S\sim P_{Sh}(S)}\left[[v_p(S+i) - v_p(S-i)]_+\right]$$

$$\varphi_i^{Bz} := \mathbb{E}_{S\sim P_{Bz}(S)}\left[[v_p(S+i) - v_p(S-i)]_+\right]$$

where $P_{Sh}(S)$ is the Shapley distribution $P_{Sh}(S) \propto \frac{1}{d+1}\binom{d}{|S|}^{-1}$ and the Banzhaf distribution is the same as the Bernoulli distribution $P_{Bz}(S) \propto p^{|S|}(1-p)^{d-|S|}$. In our experiments, we set both $p = 50\%$ as in the original Banzhaf value, but also $p = 10\%$ to consider smaller sets of documents. $[\cdot]_+$ is used to denote component-wise positive part (ReLU) which we use to take the positive part of the difference of the two probability vectors. By using these formulations on v_p instead of v_ℓ, we can use standard decoding techniques like argmax decoding or K-beam search which generate K-sparse approximations of the true v_p. These zero-probability entries in v_p will lead to $-\infty$'s in the v_ℓ approximation, which cannot be easily handled by the usual Shapley value.

3.3 Extension to Hierarchical Inputs

Leveraging natural text's intrinsic hierarchy, our method uses the structure of the retrieved passages to explain from the passage level to the sentence level to the word level. Unlike the original Shapley which treats each token as completely symmetric, no matter which document or sentence it came from, hierarchical Shapley ensures that the influence of a passage is distributed amongst its sentences and that the influence of a sentence is distributed amongst its words. While prior work (Jin et al., 2020; Chen et al., 2020) explored similar hierarchical extensions, they have only addressed binary hierarchies, lacking the support for more general structures. Instead, we support permutation sampling from a three-tiered hierarchy to calculate the Owen-Winter value. This replaces the Shapley distribution induced by sampling one ran-

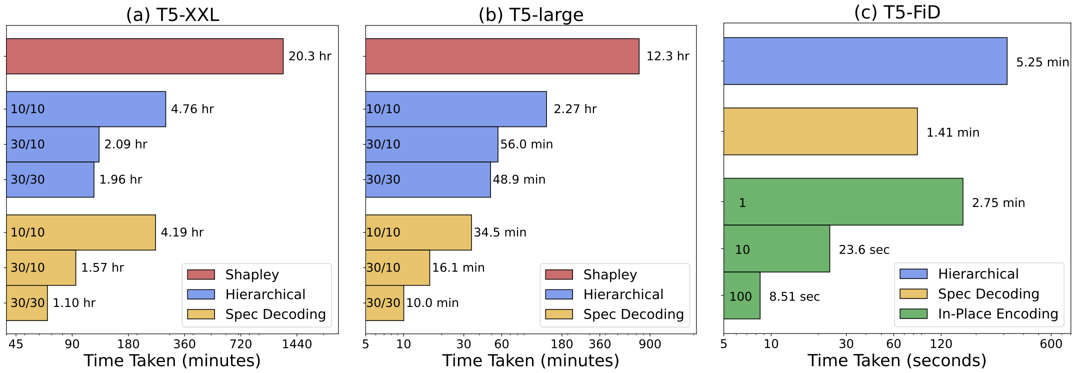


Figure 4: (a, b) TextGenSHAP speed benchmark results at the token level on T5-XXL and T5-large. (c) TextGenSHAP speed benchmark results at the document level on T5-FiD. Red is the original Shapley value with permutation sampling. Blue is the hierarchical Shapley value with hierarchical permutation sampling with thresholds in $\{10\%, 30\%\}$. Yellow is the hierarchical Shapley value with speculative decoding. Green is the hierarchical Shapley value with in-place encoding with various sizes $\{1, 10, 100\}$ for the decoding batch size (DBS).

dom permutation with the Owen-Winter distribution induced by sampling a hierarchy of random permutations (Owen, 1977; Winter, 2002).

4 TextGenSHAP: Faster Explanations

Input Hierarchy We leverage the hierarchical structure within natural text to also reduce the time complexity required for model explanation, following the theoretical foundations of the Owen-Winter value. We first break each long input document into its passages and measure the Shapley value of each passage, allowing us to select only those passages more important than some threshold to continue on to the sentence level and then word level.[2] In our experiments, we consider thresholds of both 10% and 30% for the required importance for both paragraph and sentence level. This enables us to not waste computational effort on tokens which do not warrant further investigation. A full description is available in Algorithm 1.

Speculative Decoding Another major improvement in explanation speed is gained through utilizing speculative decoding similar to what was recently explored in Miao et al. (2023); Leviathan et al. (2023). Speculative decoding, similar to branch prediction in computer architectures, uses one or multiple guesses of the generated output sequence to reduce the time taken up by repeatedly applying the decoder model. Since mistakes are corrected by reapplying the decoder model, exact output probabilities are ultimately calculated, but speed is only enhanced when guessing correctly.

Fig. 3 depicts our speculative decoding approach which is tailored to the explainability application. Existing approaches can only speculate 2-10 tokens ahead and face relatively high error rates; however, our approach speculates the entire output sequence with lower error rates. While existing methods are designed for model inference which only gets 'one guess' at the full generated output, our application is perturbation-based resampling methods like the Shapley value. This means many of our generated outputs will be similar and allows us to gradually construct a bank of speculative outputs (using a tree structure) due to the redundancies of generating similar outputs. In our experiments, we verify that a large amount of total computation can be saved by speculatively decoding full outputs rather than sequentially running the decoder model.

Specifically, for each new sample (mask) coming from the Shapley distribution, we first verify (Fig. 3b) whether the argmax decoding exists or not within our speculative decoding calculation (if so we are already done with this sample). If not, then we need to generate the new candidate answer using autoregressive decoding. Afterwards, we graft the new answer to the existing causal decoding tree so we can generate this response in the future, making sure to update the causal attention matrix in order to respect the graph structure of the decoding tree (Fig. 3d). In all experiments, we use greedy decoding consistent with prior work on open-domain QA (Izacard and Grave, 2021b; Liu et al., 2023). However, we emphasize that the

[2] Besides the paragraph-sentence-word hierarchy we consider here, other hierarchies could be better suited to other applications such as structured documents, conversation agents, or code generation.

speculative decoding tree can further support other popular sampling methods like beam search and nucleus generation (top-K and top-P) (Sina et al., 2021; Holtzman et al., 2020).

5 Architecture-Specific Accelerations

In this section, we include further details about the architectural implementations which we used to achieve the best wall-clock performance. The first section is about the widely adopted Flash Attention (Dao et al., 2022) which is used for all experiments within this paper and is what allowed for tens of thousands of tokens to be run on a single GPU. The second and third sections are additional tricks specific for the T5-FiD architecture, focusing on efficiency under the massive context sizes pursued by the long-document and open-domain communities.

Flash Attention To better address the challenges for long inputs, especially with limited compute resources, we follow recent adoptions of the Flash Attention mechanism (Dao et al., 2022) to improve both the memory efficiency and the runtime performance of LMs. Such approaches compute the attention matrix with the memory requirement scaling linearly with input size $\mathcal{O}(N)$ instead of quadratically $\mathcal{O}(N^2)$ (Rabe and Staats, 2022; Dao, 2023).

Block Sparse Attention We make a connection between Flash Attention and recent developments in long-document architectures (Izacard and Grave, 2021b; Beltagy et al., 2020; Ding et al., 2023) by using block sparse attention matrices for handling long inputs. Accordingly, we reformulate the original FiD to also incorporate a block sparse implementation of Flash Attention, still respecting the hardware-aware block sizes. These techniques demonstrate our method can be useful at immense context sizes, which is of heightened necessity as the context lengths of modern LLM architectures continue to grow.

In-Place Resampling We exploit the unique structure of chunking-based encoder-decoder models like FiD to get speedups significantly faster than previously attainable. In particular, we compute the encoder feature matrix just once while generating the entire explanation for a single example. Due to the independence of chunked input fragments, we only need to adjust the encoder-decoder cross-attention mechanism to enable resampling with different document subsets. Reducing the memory overhead not only reduces the computation time for re-encoding features, but allows for quicker memory accesses and larger throughput via a 'decoding batch size' which generates multiple outputs for a single input context. Increasing the decoding batch size enables much more hardware-efficient decoding (iterating through hundreds of permutation samples in only seconds on a single GPU).

6 Experimental Results

Datasets We focus on publicly-available datasets for the task of open-domain or long-document question answering: Natural Questions (NQ) (Kwiatkowski et al., 2019) and MIRACL (English subset) (Zhang et al., 2022b). We follow NQ as redesigned for open-domain question answering following (Lee et al., 2019; Karpukhin et al., 2020) called NQ-Open. In this setting, answers must be found from within all of Wikipedia, rather than a single Wikipedia page. The original NQ dataset provides short text answers and passages are rated as relevant so long as they contain the ground-truth answer. MIRACL is instead designed for information retrieval and for each query it provides binary relevance ground-truth for the ten most related passages in the corpus.[3]

Models For passage ranking of the corpus (retriever model) we use the recent Contriever (Izacard et al., 2022a) architecture following LitM. For question answering (reader models) we use different members of the T5 family (Raffel et al., 2020). We use the available flan tuned models at the large and XXL sizes ('T5-large' and 'T5-XXL') (Chung et al., 2022) and the fine-tuned T5 large model from FiD ('T5-FiD') (Izacard and Grave, 2021b).

6.1 TextGenSHAP Speed Benchmarking

We present benchmarks demonstrating the improved speed of TextGenSHAP. A single A100 40GB GPU is used for benchmarking all experiments. We note that our method would further benefit from parallelism across multiple GPUs.

First, we evaluate the standard Shapley value, which provides detailed token-level explanations using our Algorithm 1. In Fig. 4, we benchmark with 100 sampled permutations and 10 documents from the LitM setting for both T5-XXL and T5-large. We observe that the standard Shapley value estimation requires a prohibitive 12-20 hours per

[3] In this work, 'documents' and 'passages' are used synonymously as the unit of retrieval for both NQ and MIRACL.

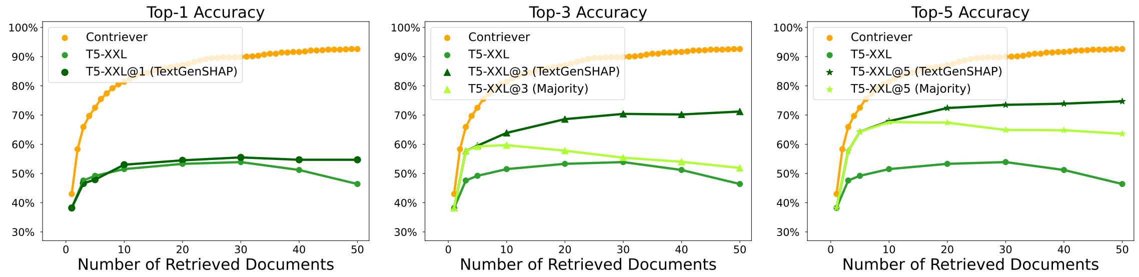


Figure 5: Top-K Accuracy for K=1, 3, 5 on the Natural Questions dataset for the original model, majority vote baseline, and explanation-based resorting method (TextGenSHAP). The upper bound of the retrieval score is also included (Contriever).

sample and show that our proposed hierarchical sampling algorithm significantly reduces this time. With the integration of speculative decoding, we can achieve an even more significant reduction in computation time, bringing computation time to nearly an hour or often faster. We note that additional speedups can be achieved in real-world settings by just sampling fewer permutations. In Appendix B.1, we show that much fewer than 100 permutation samples can suffice for accuracy gains. When using only 10 permutation samples, TextGenSHAP reduces the time for the T5-XXL model from about two hours to five minutes. We additionally benchmark the T5-FiD model accelerated with its architecture specific modifications as seen in Fig. 4c. We take document-level explanations from multiple minutes to less than ten seconds, enabling real-time improvements for document retrieval applications (see Sec. 6.4).

6.2 Visualizing Interpretations

We provide an example visualization in Fig. 2 to demonstrate the hierarchy enabled by TextGenSHAP. We observe the model consistently grounds its answer to the first document, which indeed contains the true answer to the example question. We also find that our hierarchical Shapley scores are effective for isolating important tokens from within contexts of thousands of tokens. We present further visualizations in Appendix D, and provide an interactive visualization hosted here.

6.3 Improved Question Answering

We study using TextGenSHAP to refine long information contexts. Following the recommendations in (Liu et al., 2023), we refine the model's available documents before reaching a final answer. We evaluate top-K accuracy for small values of K, narrowing the existing gap between the retriever's recall and the reader's accuracy, which highlights the importance of providing a diverse set of candidate answers. Fig. 5 illustrates the accuracy improvements achieved by the redistilled model compared to the majority voting baseline. TextGenSHAP significantly outperforms the baseline model, and further surpasses the majority voting baseline's AUC scores in Table 1.

Table 1: AUC for the accuracy curves in Fig. 5 on NQ.

	K=1	K=3	K=5
Baseline	50.54	–	–
Majority Vote	32.90	55.19	63.88
TextGenSHAP	**52.72**	**66.16**	**69.57**

6.4 Improved Retrieval

We show the value of the proposed explanation scores in TextGenSHAP for the use case of document retrieval for open-domain QA. We propose improving the retriever by enhancing the recall of the modified retriever model using reranked passages according to their explanation scores.

Table 2: AUC for the recall curves on both the NQ dataset and MIRACL dataset.

	Natural Questions	MIRACL (Original)	MIRACL (Pseudo)
Baseline	84.23	80.18	84.53
TextGenSHAP	88.53	77.33	86.43
TextGenBANZ	88.56	78.19	86.17
TextGenBANZ-10	88.74	82.38	86.53
Attention[4]	88.35	78.27	84.30

Table 2 shows substantial recall improvement on the NQ dataset, with all three of our proposed explanation methods exhibiting similar performance improvements compared to the baseline retriever

[4] Attention follows the best hyperparameters for aggregation found in (Izacard and Grave, 2021a)

model. Comparing to an existing method of reranking based on distilling attention scores (Izacard and Grave, 2021a), we see our approaches are able to achieve comparable performance.

Less pronounced improvements on the more challenging MIRACL dataset may primarily be due to its sparser label information, only providing labels for ten of the millions of available passages. We verify this claim by extending the label information using pseudo-labels. Specifically, we take all relevant passages according to the MIRACL labels and ask T5-XXL to give a short answer according to that passage alone. We then leverage this set of candidate answers to evaluate passage relevance similar to the NQ dataset. In the last column of Table 2, we see this not only improves the overall recall, but disproportionately boosts the success of TextGenSHAP, highlighting its ability to discover relevant passages missed by existing retrieval methods.

6.5 LLM Hallucinations

Addressing LLM hallucinations is of growing importance given modern LLM usage. Retrieval-augmented generation is seen as one effective solution (Shuster et al., 2021; Gao et al., 2023), making TextGenSHAP well posed to be able to identify and eliminate hallucinations from LLMs. By providing explainable results, our method enables human-in-the-loop approaches to further tackle the problem of hallucination. See Appendix E for further details.

6.6 Dataset Repair

As discussed in Sec. 6.4, our method can not only identify documents which are often underexplored by existing approaches, allowing for greater diversity in data collection, but also is able to localize critical information within extensive documents. Accordingly, we suggest that our method could enhance dataset construction pipelines by significantly reducing the burden of human annotation. Examples of this capability on the MIRACL dataset is provided in Appendix F .

7 Conclusion

In this paper, we introduce TextGenSHAP for enhancing the Shapley value, a trusted explainability method, to address the challenges in modern NLP applications featuring long inputs, large model sizes, and text generation. We introduce modifications to adapt the Shapley value for hierarchically-structured input text and autoregressively-decoded output generations, drawing on insights from the game theory literature to support their theoretical motivations. Additionally, we incorporate multiple transformer-specific architecture modifications which significantly accelerate explanation generation. Our approach not only speeds up Shapley value computation for generated text but also demonstrates its effectiveness in improving performance at challenging question-answering tasks. We expect that such explanation methods will continue to find broad applicability in a variety of LLM use cases.

8 Limitations

The primary goal of this work is to introduce a variation of the Shapley value for generative LLMs. Although the definition is well-motivated and the experiments show significant improvement in wall-clock time, many applications still have the potential to face the concern that the generated explanations could be too costly to incorporate. Models which are distributed over multiple GPUs or TPUs, are left unexplored in the current work, possibly requiring further verification to guarantee speedups of the hardware-specific modifications we make. Verification of speed and accuracy improvements over a larger range of tasks and architectures is left to future work, and evaluation against existing post-hoc techniques (IG and LIME) is limited due to the infeasibility of comparison.

References

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Ines Arous, Ljiljana Dolamic, Jie Yang, Akansha Bhardwaj, Giuseppe Cuccu, and Philippe Cudré-Mauroux. 2021. Marta: Leveraging human rationales for explainable text classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(7):5868–5876.

John F. III Banzhaf. 1965. *Weighted Voting Doesn't Work: A Mathematical Analysis*, volume 19, pages 317–344.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Amanda Bertsch, Uri Alon, Graham Neubig, and Matthew R. Gormley. 2023. Unlimiformer: Long-range transformers with unlimited length input.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Aydar Bulatov, Yuri Kuratov, and Mikhail S. Burtsev. 2023. Scaling transformer to 1m tokens and beyond with rmt.

Aydar Bulatov, Yury Kuratov, and Mikhail Burtsev. 2022. Recurrent memory transformer. In *Advances in Neural Information Processing Systems*, volume 35, pages 11079–11091. Curran Associates, Inc.

Aaron Chan, Maziar Sanjabi, Lambert Mathias, Liang Tan, Shaoliang Nie, Xiaochang Peng, Xiang Ren, and Hamed Firooz. 2022. UNIREX: A unified learning framework for language model rationale extraction. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 2867–2889. PMLR.

Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. 2020. Generating hierarchical explanations on text classification via feature interaction detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5578–5593, Online. Association for Computational Linguistics.

Jianbo Chen, Le Song, Martin J. Wainwright, and Michael I. Jordan. 2019. L-shapley and c-shapley: Efficient model interpretation for structured data. In *International Conference on Learning Representations*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Ian Covert, Scott Lundberg, and Su-In Lee. 2021. Explaining by removing: A unified framework for model explanation. *Journal of Machine Learning Research*, 22(209):1–90.

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pages 16344–16359.

Jiayu Ding, Shuming Ma, Li Dong, Xingxing Zhang, Shaohan Huang, Wenhui Wang, Nanning Zheng, and Furu Wei. 2023. Longnet: Scaling transformers to 1,000,000,000 tokens.

Nouha Dziri, Sivan Milton, Mo Yu, Osmar Zaiane, and Siva Reddy. 2022. On the origin of hallucinations in conversational models: Is it the datasets or the models? In *Proceedings of the 2022 Conference of* the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5271–5285, Seattle, United States. Association for Computational Linguistics.

Dan S. Felsenthal and Moshé Machover. 1998. *The Measurement of Voting Power: Theory and Practice, Problems and Paradoxes*. Edward Elgar Publishing, Cheltenham, UK.

Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade v2: Sparse lexical and expansion model for information retrieval.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Amirata Ghorbani, Abubakar Abid, and url=https://ojs.aaai.org/index.php/AAAI/article/view/4252 DOI=10.1609/aaai.v33i01.33013681 number=01 journal=Proceedings of the AAAI Conference on Artificial Intelligence James Zou, volume=33. 2019. Interpretation of neural networks is fragile. pages 3681–3688.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Jie Huang, Wei Ping, Peng Xu, Mohammad Shoeybi, Kevin Chen-Chuan Chang, and Bryan Catanzaro. 2023a. Raven: In-context learning with retrieval augmented encoder-decoder language models. *arXiv preprint arXiv:2308.07922*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023b. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022a. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Gautier Izacard and Edouard Grave. 2021a. Distilling knowledge from reader to retriever for question answering. In *International Conference on Learning Representations*.

Gautier Izacard and Edouard Grave. 2021b. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880. Association for Computational Linguistics.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022b. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*.

Alon Jacovi and Yoav Goldberg. 2021. Aligning faithful interpretations with their social attribution. *Transactions of the Association for Computational Linguistics*, 9:294–310.

Alon Jacovi, Swabha Swayamdipta, Shauli Ravfogel, Yanai Elazar, Yejin Choi, and Yoav Goldberg. 2021. Contrastive explanations for model interpretability. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1597–1611, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Neil Jethani, Mukund Sudarshan, Ian Connick Covert, Su-In Lee, and Rajesh Ranganath. 2022. FastSHAP: Real-time shapley value estimation. In *International Conference on Learning Representations*.

Xisen Jin, Zhongyu Wei, Junyi Du, Xiangyang Xue, and Xiang Ren. 2020. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. In *International Conference on Learning Representations*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Brihi Joshi, Aaron Chan, Ziyi Liu, Shaoliang Nie, Maziar Sanjabi, Hamed Firooz, and Xiang Ren. 2022. ER-test: Evaluating explanation regularization methods for language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3315–3336, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of* *the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Meinard Kuhlmann. 2023. Quantum Field Theory. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*, Summer 2023 edition. Metaphysics Research Lab, Stanford University.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. 2023. Measuring faithfulness in chain-of-thought reasoning.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning*.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2023a. Towards faithful model explanation in nlp: A survey.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023b. Faithful chain-of-thought reasoning. pages 305–329, Nusa Dua, Bali. Association for Computational Linguistics.

Xueguang Ma, Kai Sun, Ronak Pradeep, and Jimmy Lin. 2021. A replication study of dense passage retriever.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Generation-augmented retrieval for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4089–4100, Online. Association for Computational Linguistics.

Xupeng Miao, Gabriele Oliaro, Zhihao Zhang, Xinhao Cheng, Zeyu Wang, Rae Ying Yee Wong, Alan Zhu, Lijie Yang, Xiaoxiang Shi, Chunan Shi, Zhuoming Chen, Daiyaan Arfeen, Reyna Abhyankar, and Zhihao Jia. 2023. Specinfer: Accelerating generative large language model serving with speculative inference and token tree verification.

Rory Mitchell, Joshua Cooper, Eibe Frank, and Geoffrey Holmes. 2022. Sampling permutations for shapley value estimation. *Journal of Machine Learning Research*, 23(43):1–46.

Edoardo Mosca, Ferenc Szigeti, Stella Tragianni, Daniel Gallagher, and Georg Groh. 2022. SHAP-based explanation methods: A review for NLP interpretability. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4593–4603, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Guilliermo Owen. 1977. Values of games with a priori unions. In *Mathematical Economics and Game Theory*, pages 76–88, Berlin, Heidelberg. Springer Berlin Heidelberg.

L. S. Penrose. 1946. The elementary statistics of majority voting. 109(1):53 – 57.

Markus N. Rabe and Charles Staats. 2022. Self-attention does not need $o(n^2)$ memory.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *arXiv preprint arXiv:2302.00083*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. “why should i trust you?” explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead.

Gabriele Sarti, Nils Feldhus, Ludwig Sickert, and Oskar van der Wal. 2023. Inseq: An interpretability toolkit for sequence generation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 421–435, Toronto, Canada.

L. S. Shapley. 1953. *A Value for n-Person Games*, volume 2, pages 307–318. Princeton University Press, Princeton.

L. S. Shapley and Martin Shubik. 1954. A method for evaluating the distribution of power in a committee system. 48(3).

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zarriess Sina, Henrik Voigt, and Simeon Schüz. 2021. Decoding methods in neural language generation: A survey. *Information*, 12(9).

Joe Stacey, Yonatan Belinkov, and Marek Rei. 2022. Supervising model attention with human explanations for robust natural language inference. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11349–11357.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Junlin Wang, Jens Tuyls, Eric Wallace, and Sameer Singh. 2020. Gradient-based analysis of NLP models is manipulable. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 247–258, Online. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Eyal Winter. 2002. Chapter 53 the shapley value. volume 3 of *Handbook of Game Theory with Economic Applications*, pages 2025–2054. Elsevier.

Yuhuai Wu, Markus Norman Rabe, DeLesley Hutchins, and Christian Szegedy. 2022. Memorizing transformers. In *International Conference on Learning Representations*.

Kayo Yin and Graham Neubig. 2022. Interpreting language models with contrastive explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 184–198, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hang Zhang, Yeyun Gong, Yelong Shen, Weisheng Li, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2022a. Poolingformer: Long document modeling with pooling attention.

Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. 2022b. Making a MIRACL: Multilingual information retrieval across a continuum of languages. *arXiv:2210.09984*.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2023. Explainability for large language models: A survey.

Yiming Zheng, Serena Booth, Julie Shah, and Yilun Zhou. 2022. The irrationality of neural rationale models. In *Proceedings of the 2nd Workshop on Trustworthy Natural Language Processing (TrustNLP 2022)*, pages 64–73, Seattle, U.S.A. Association for Computational Linguistics.

A LitM Reverification

We utilize many experiments to understand the degree of the claims from (Liu et al., 2023). In particular, we further verify how dependent it is on the semi-synthetic distribution introduced by the authors therein. There are a few major assumptions made in this semi-synthetic distribution (of planting a single document amongst a set of distractor documents) which may not always hold up in practical scenarios. First, the number of documents which are retrieved in real-world systems containing the true answer will not be exactly equal to one. Second, the order and relevancy of distractor documents may vary by retrieval system used and by documents within a corpus.

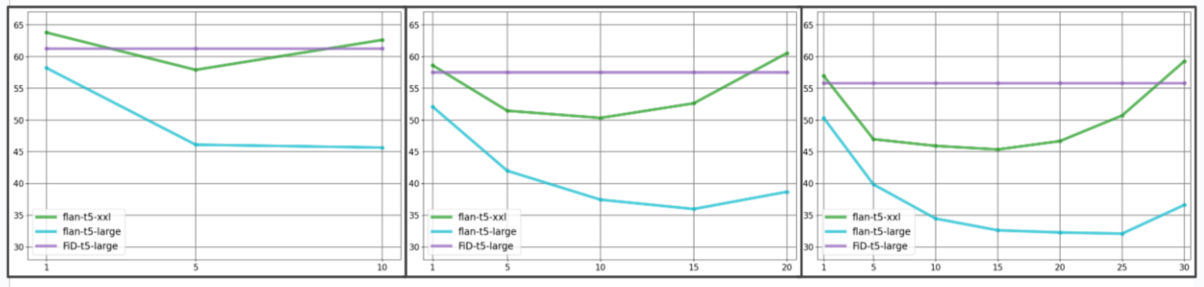


Figure 6: Reproducing the 'Lost in the Middle' phenomena from (Liu et al., 2023) in the proposed setup.

For all reader models we utilize, we verify the hypothesis from (Liu et al., 2023) on the effect of document position on model performance. In Fig. 6, we indeed see for the models trained in the typical way like T5-large and T5-XXL, we indeed reverify the hypothesis of LitM which shows a degradation in model performance whenever the true answer is placed towards the center of a very long context window. We additionally compare the performance of the permutation invariant T5-FiD model. Here, we consequently see that the model architecture trained to perform the long-document question answering task is able to increase the performance over the original T5-large model. In fact, we see that for some parts of the LitM curve, that the smaller T5-FiD model is able to outperform the much larger T5-XXL model.

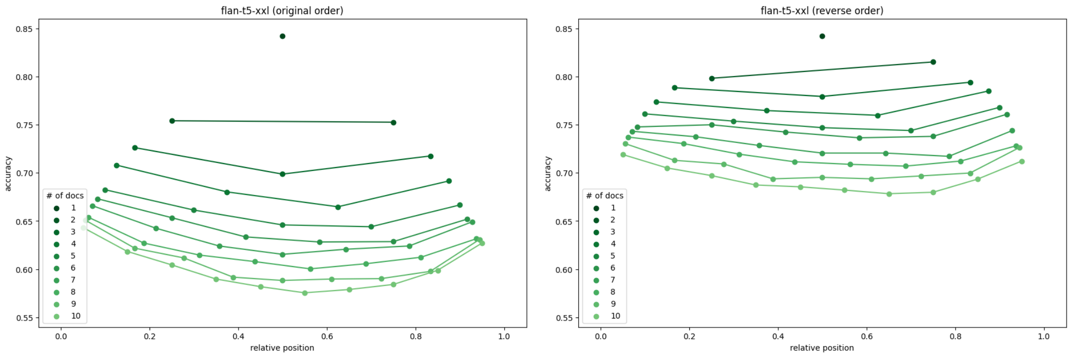


Figure 7: Accuracy vs. the document position. We demonstrate that the 'lost in the middle phenomena' (Liu et al., 2023) can be mitigated with inclusion of less relevant, distractor, documents.

To further prod the findings from (Liu et al., 2023), we investigate how changing the distractor documents in the context will alter the decision making with long context. Instead of taking the top 10 most relevant passages to serve as the distractor documents (as done (Liu et al., 2023)), we look at taking some less relevant retrieved passages by reversing the order of the top-K selected. Fig. 7 shows that making this change to the semi-synthetic setup indeed reduces the depth of the bowl-shaped curve.

B Experiment Details

B.1 Models and Datasets

Datasets Natural Questions (NQ) (Kwiatkowski et al., 2019) is a dataset originally designed for long-document question answering, where both a relevant passage and a final answer must be selected from a single Wikipedia page. NQ is redesigned for open-domain question answering following (Lee et al., 2019; Karpukhin et al., 2020) which convert Wikipedia into a corpus of passages instead of pages, and only require giving a final answer which can be found amongst said passages. The original NQ dataset provides short text answers and passages are rated as relevant so long as they contain the ground-truth answer.

MIRACL (Zhang et al., 2022b). is a dataset designed for information retrieval over Wikipedia passages. Using an existing information retrieval score, the dataset selected the ten most relevant passages the corpus and labeled each as either relevant or irrelevant to the question at hand. Relevance judgements are made by a human annotator who decides whether the passage information is sufficient to answer the given question; however, they are not required to justify or describe the answer as part of the label. Accordingly, only a handful of passages have ground-truth single-judgement label information. This constitutes a much sparser signal than the NQ dataset which allows for any passage which contains the ground-truth text answer to be deemed as relevant. It is for this reason we generate psuedolabels based off of the relevant MIRACL passages to reevaluate MIRACL passages using the same criteria as NQ. In this work, we only focus on the subsest of MIRACL which uses English queries and English passages.

Models We follow the standard two-stage pipeline of ODQA, first using a retriever model to select a subset of relevant passages from a massive corpus and second using a reader model to extract the question's answer from the subset of relevant passages.

For passage ranking of the corpus (retriever model), we use the recent Contriever (Izacard et al., 2022a) architecture following LitM , using FAISS to index the embeddings (Johnson et al., 2019). For question answering (reader model), we use different members of the T5 family (Raffel et al., 2020). We use the available flan-tuned models at the large and XXL sizes ('T5-large' and 'T5-XXL') (Chung et al., 2022) and the fine-tuned T5 large model from FiD ('T5-FiD') (Izacard and Grave, 2021b). Specifically, these correspond to `flan-t5-large` and `flan-t5-xxl` available from (Chung et al., 2022) which are originally trained on contexts of length 512. T5-FiD corresponds to `nq_reader_large` from (Izacard and Grave, 2021b) which is originally trained on context lengths of one hundred passages retrieved from their co-trained retriever. Despite the sizes of training context lengths, it is common to apply such models beyond their originally trained context lengths when applied to the task of long-document question answering (Liu et al., 2023) (which is feasible due to the relative position bias implemented within T5).

B.2 Additional Results

Here we provide the additional results for various values different values of the number of permutations used to generate explanations before evaluating. Because this is the main knob for sampling based algorithms to trade between estimation accuracy and time complexity, we calculate the AUC metrics of our target application across all levels of permutations to show the different effects. We see that even in as few as ten permutations we are getting multiple points of recall AUC in the end-to-end information retrieval system.

Table 3: AUC for 3 permutations.

	Natural Questions	MIRACL (Original)	MIRACL (Pseudo)
Baseline	84.23	80.18	84.53
TextGenSHAP	86.01	69.58	84.71
TextGenBANZ	85.76	72.84	84.80
TextGenBANZ-10	87.53	79.08	85.40

Table 4: AUC for 10 permutations.

	Natural Questions	MIRACL (Original)	MIRACL (Pseudo)
Baseline	84.23	80.18	84.53
TextGenSHAP	87.50	74.52	85.39
TextGenBANZ	87.86	75.65	85.71
TextGenBANZ-10	88.61	81.39	86.27

Table 5: AUC for 30 permutations.

	Natural Questions	MIRACL (Original)	MIRACL (Pseudo)
Baseline	84.23	80.18	84.53
TextGenSHAP	88.31	76.71	85.97
TextGenBANZ	88.51	76.88	86.27
TextGenBANZ-10	88.77	82.15	86.60

Table 6: AUC for 100 permutations.

	Natural Questions	MIRACL (Original)	MIRACL (Pseudo)
Baseline	84.23	80.18	84.53
TextGenSHAP	88.53	77.33	86.43
TextGenBANZ	88.56	78.19	86.17
TextGenBANZ-10	88.74	82.38	86.53

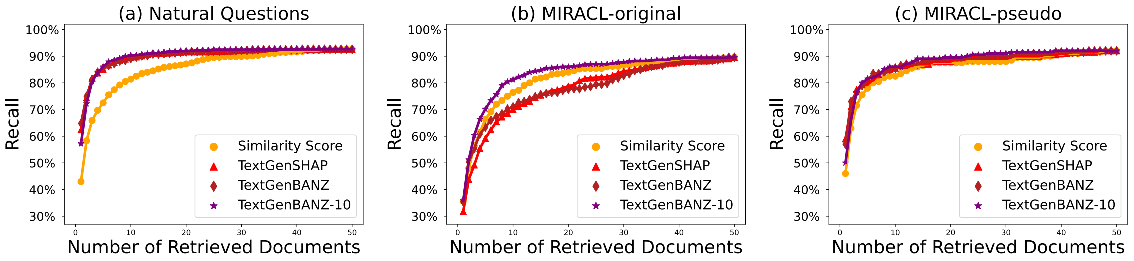


Figure 8: Recall improvements via resorting the retrieved documents using different methods (a) Natural Questions (b) MIRACL with original labels (c) MIRACL with pseudo labels

C Further Details on the Shapley Value

As a reminder, we consider a language model $F : [V]^d \to [0,1]^{[V]^m}$ and we take $f(x, S) := F(\boldsymbol{x} \odot \boldsymbol{s} + \boldsymbol{p} \odot (1 - \boldsymbol{s}))$ to define a masked language model $f : [V]^d \times \{0,1\}^d \to [0,1]^{[V]^m}$ where the inputs, input masks, and outputs are $\boldsymbol{x} \in [V]^d$, $\boldsymbol{s} \in \{0,1\}^d$, and $\boldsymbol{y} \in [V]^m$, respectively. We consider a value function $v : \mathcal{P}([d]) \to \mathbb{R}^M$ for $M = V^m$, and consider the choices of value function as the log-probabilities or probabilities: $v_\ell(S) := \log(f(x, 1_S))$ and $v_p(S) := f(x, 1_S)$. Please refer back to the notation section in the main text for full details if necessary.

C.1 Shapley Value

The Shapley value is a long-existing solution concept from the game theory literature, originally designed to correctly attribute the value of each individual player within a cooperative game of forming a coalition (Shapley, 1953). In recent years, this solution concept has been repurposed towards the goal of explaining black-box machine learning models, treating each individual feature as a player and dividing up the prediction output correctly between the features (Lundberg and Lee, 2017). Between this time, however, many further advancements in the game theory literature building off of the seminal work by Shapley have continued to progress. Herein, we focus on a few such extensions of the original Shapley value as we apply them to our particular structured data of text-to-text generation models.

The first such advancement occurred only shortly after the original Shapley value's conception; the Shapley-Shubik power index is a reformulation of the original Shapley value instead designed for voting games (Shapley and Shubik, 1954). Here, the Shapley-Shubik value measures the amount of power or influence each voter has to influence the outcome of the vote. Also in the category of voting games, the Penrose-Banzhaf index (or more commonly Banzhaf power index) was first discovered by Penrose (Penrose, 1946) and was later independently discovered by Banzhaf (Banzhaf, 1965). Even now, both Banzhaf and Shapley-Shubik remain the two well-respected pillars for how to effectively evaluate the structure of a voting game.

Along the direction of further extensions to the Shapley value, Owen years later extended the Shapley value to additional deal with a two-level hierarchical structure (Owen, 1977). In particular, one can imagine that players form coalitions within an organization but moreover that organizations themselves form coalitions with one another. The value can further be defined for multi-level hierarchical structures and is sometimes called the Owen-Winter value (Winter, 2002). The corresponding extension to the Banzhaf value is instead usually considered more straightforward and is also referred to as the Banzhaf value. In this work, we use a combination of all listed approaches to be able to apply SHAP-style (Lundberg and Lee, 2017) explanations of machine learning algorithms in the case of sequence-to-sequence transformer models, adapting to the hierarchical structure of input text and the autoregressive structure of output text.

The Shapley value is commonly formulated as a uniform expectation over permutations, which lends itself to approximation via permutation sampling:

$$\varphi_i = \mathbb{E}_\pi \Big[v_\ell(S_{\pi,i}+i) - v_\ell(S_{\pi,i}-i) \Big] = \frac{1}{|\mathcal{S}_d|} \sum_{\pi \in \mathcal{S}_d} \Big\{ v_\ell(S_{\pi,i}+i) - v_\ell(S_{\pi,i}-i) \Big\} \tag{2}$$

where $\pi \in \mathcal{S}_d := \{\pi : [d] \to [d] : \pi \text{ is bijective}\}$ is the set of permutations of size d and the expectation is computed over the uniform distribution of permutations. In other words, π represents a random order of the features (tokens) and $S_{\pi,i} := \{j \in [d] : \pi(j) < \pi(i)\}$ is the set of elements which precede i in the order defined by π. Hence, $S_{\pi,i} + i = \{j \in [d] : \pi(j) \le \pi(i)\}$ and $S_{\pi,i} - i = S_{\pi,i} = \{j \in [d] : \pi(j) < \pi(i)\}$.

We can equally well write the Shapley value as the average over the induced distribution on the subsets $S \in \mathcal{P}([d])$:

$$\varphi_i = \mathbb{E}_{S \sim P_{Sh}(S)} \Big[v_\ell(S+i) - v_\ell(S-i) \Big] = \sum_{S \subseteq [d]} \frac{1}{(d+1)\binom{d}{|S|}} \cdot \Big\{ v_\ell(S+i) - v_\ell(S-i) \Big\} \tag{3}$$

where $P_{Sh}(S)$ is the Shapley distribution $P_{Sh}(S) \propto \frac{1}{d+1}\binom{d}{|S|}^{-1}$

Because all such definitions of this solution concept involve at least an exponential amount of terms to compute exactly, the standard approach in the literature is to use permutation sampling (Covert et al., 2021; Mitchell et al., 2022). In this work, we additionally follow the approach of permutation sampling, making adjustments as necessary to apply to hierarchical structure as described in Algorithm 1.

C.2 Shapley-Shubik

Our first important departure from the existing Shapley literature is to be able to handle the case of autoregressively decoded output sequences. All existing post-hoc explanations including attention-based, gradient-based, and perturbation-based methods cannot be directly applied to text generations. Further details on these shortcomings of existing works are further described in Section 2. In such applications to text generation when they do exist, are done autoregressively, explaining each of the output tokens individually sometimes even without regard for the decoded outputs occurring prior to each autoregressive output. Not only does this pose a serious visualization challenge as decoded outputs get longer and longer in the era of LLMs, but also the correlations of explanations between adjacent output tokens are often left improperly handled.

Algorithm 1 Pseudo-code for efficient hierarchical Shapley computation

```
**Input**: data sample $x \in [V]^d$, masked text generation model $f : [V]^d \times \{0,1\}^d \to [V]^m$, number of passages $p \in \mathbb{N}$, number of tokens $d \in \mathbb{N}$, hierarchical partition of tokens $P = (S_1, \dots, S_p)$
**Parameters**: hierarchy threshold $\tau$, number of samples $T$
**Output**: computed Shapley values at document level $\{\varphi_k\}_{k\in[p]}$ and token level $\{\varphi_{k,i}\}_{k\in\mathcal{I},i\in S_k}$

function RandPerm($N$)
    return {random permutation of $N$}
function OneShapleyPath($f$, $P$, $\mathcal{I}$, $\varphi_k$, $\varphi_{k,i}$)
    $\pi \leftarrow$ RandPerm($p$),    $S \leftarrow \emptyset$,    $\text{text}_{\text{curr}} \leftarrow$ “ ”        ▷ Initialize the loop
    for $k = 1 : p$ do
        if $k \notin \mathcal{I}$ then        ▷ Case 1: Add all of the unimportant document's tokens to $S$
            $S \leftarrow S \cup S_{\pi(k)}$        ▷ Add the entire document
            if $f(x; 1_S) \neq \text{text}_{\text{curr}}$ then
                Increment the count of text $f(x; 1_S)$ in $\varphi_{\pi(k)}$ by one
                $\text{text}_{\text{curr}} \leftarrow f(x; 1_S)$
        else        ▷ Case 2: Add the important document's tokens one by one
            $\pi_k \leftarrow$ RandPerm($S_k$)        ▷ Random order of the tokens within the document
            for $i \in S_k$ do        ▷ Iterate through each token in the document
                $S \leftarrow S \cup \{\pi_k(i)\}$        ▷ Add a single token
                if $f(x; 1_S) \neq \text{text}_{\text{curr}}$ then
                    Increment the count of text $f(x; 1_S)$ in $\varphi_{\pi(k),\pi_k(i)}$ by one
                    $\text{text}_{\text{curr}} \leftarrow f(x; 1_S)$

function HierarchicalShapley
    Initialize $\varphi_k \leftarrow \vec{0}$, for each $k \in [p]$
    Initialize $\varphi_{k,i} \leftarrow \vec{0}$ for each $k \in [p]$, $i \in S_k$
    for $t = 1 : T$ do
        OneShapleyPath($f, P, \emptyset, \varphi_k, \varphi_{k,i}$)        ▷ First, only sample at the document level
    $\mathcal{I} \leftarrow \{k \in [p] : \varphi_k / S \geq \tau\}$        ▷ Select the set of important documents
    for $t = 1 : T$ do
        OneShapleyPath($f, P, \mathcal{I}, \varphi_k, \varphi_{k,i}$)        ▷ Second, sample at the token level for certain documents
    return $\{\varphi_k\}_{k\in[p]}, \{\varphi_{k,i}\}_{k\in[p],i\in S_k}$
```

This challenge stems from the fact that when using autoregressive sequence-to-sequence models, the full output probability vector is never calculated. We need to utilize decoding schemes like greedy decoding, K-beam generation, or nucleus decoding to approximate the most likely parts of the output generation space. In contrast to existing post-hoc approaches, our method is able to explain the full output sequence by reformulating Shapley into the Shapley-Shubik formulation on the probability vector and yielding an explanation on the entire prediction sequence.

We define the Shapley-Shubik and Banzhaf values as :

$$\varphi_i^{Sh} := \mathbb{E}_{S\sim P_{Sh}(S)}\Big[[v_p(S+i) - v_p(S-i)]_+\Big] \quad \varphi_i^{Bz} := \mathbb{E}_{S\sim P_{Bz}(S)}\Big[[v_p(S+i) - v_p(S-i)]_+\Big] \tag{4}$$

where $P_{Sh}(S)$ is the Shapley distribution $P_{Sh}(S) \propto \frac{d-1}{\binom{d}{|S|}|S|(d-|S|)}$ and the Banzhaf distribution is the same as the Bernoulli distribution $P_{Bz}(S) \propto p^{|S|}(1-p)^{d-|S|}$.

Accordingly, our Shapley explanation will be well-defined even on the sparse probability vectors v_p which are induced by all natural decoding algorithms. It is for this reason we are able to generate explanations on the entire prediction output unlike existing SHAP approaches, handling generated text coming from distributions of a-priori unknown support.

C.3 Existing Variations for NLP Applications

In this section, we further detail existing work and the similarities and differences between the approaches taken therein.

C.3.1 Hierarchical Variants

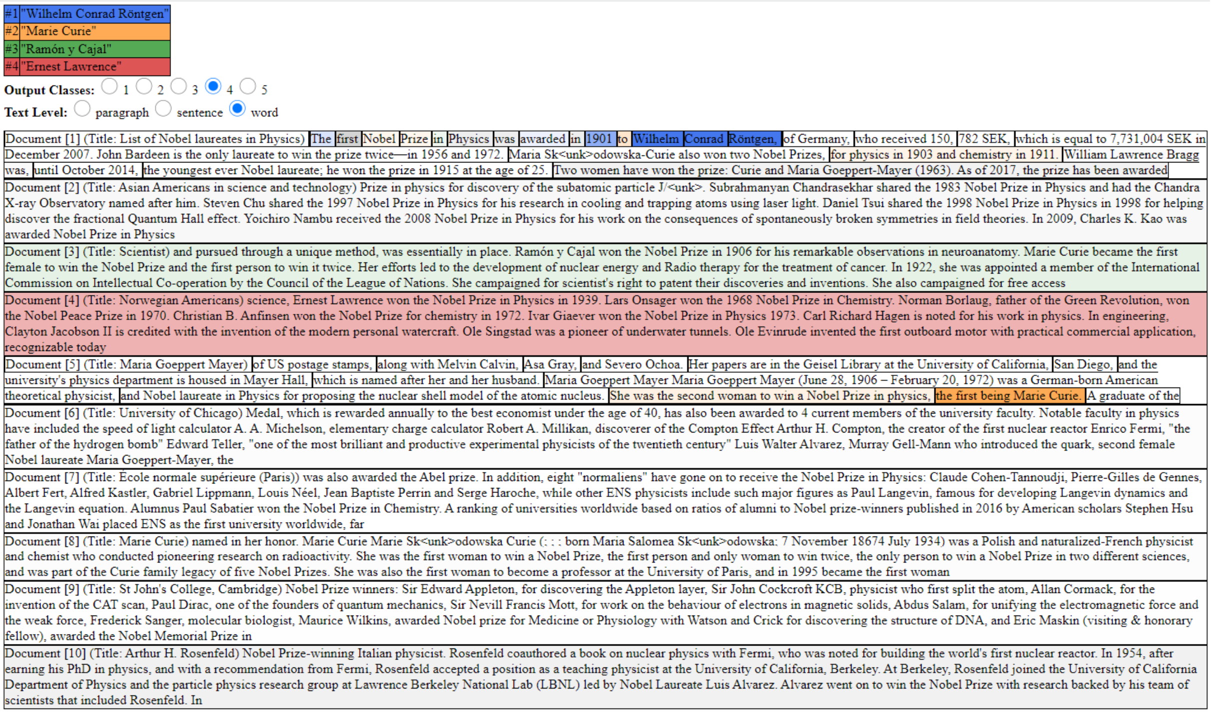
In the literature on Shapley for NLP or perturbation-based explanations for NLP, there have already been approaches leveraging the sequential and/or hierarchical structure of NLP data. In this section, we highlight the similarities and differences of existing approaches. One of the earliest approaches using structured versions of the Shapley value, (Chen et al., 2019) defines a Shapley value which can only consider coalitions with its neighbors (using linear structure for text data) meaning that word interactions will only span across adjacent phrases. This work does not explicitly leverage the further hierarchical structure of text data, but still utilizes input structure of text information. One of the earliest works using the hierarchical structure, (Jin et al., 2020), uses human-labeled grammatical hierarchies coming from the SST-2 sentiment classification dataset to assist in generating explanations. Their explanations give values to each node in the hierarchy and are done using their sampling and occlusion algorithm, similar to perturbation-based approaches from the interpretability literature. Finally, (Chen et al., 2020) automatically generates a hierarchy over the input text via a specially designed splitting algorithm. Phrases are split in binary pairs by choosing the weakest set of interacting phrases. Searching over phrase splits can be done in linear time by assuming phrases are sequential. Accordingly, all existing approaches will only apply to binary hierarchies and there are no existing approaches which can handle more complex hierarchies like the paragraph-sentence-word tiering which we consider in this work by utilizing permutation sampling on the Owen-Winter value.

C.3.2 Constrastive Variants

Additionally, there have also been more recent advancements on the output structure side for Shapley-style attributions. In the context of language modeling (text to text) applications, there is a greater need to handle the growing complexity of an explanation with respect to the language model. While many works have tried the simple reformulation of language modeling as a classification task of the first produced token, fewer works have made further progress in providing sensible explanations beyond a vector over all possible output tokens (often amongst tens of thousands of tokens or more). In particular, the main approach leveraged is that of contrastive explanations, which specifically requires a comparison between two alternative output tokens, rather than a broad explanation across them all. (Jacovi et al., 2021) applies these techniques to still the simpler case of multiclass classification, highlighting the value of contrastive explanations for NLP applications. More recently, (Yin and Neubig, 2022) applies similar techniques to the case of language modeling on the first token, using grammatical information as useful candidates for contrastive explanations. Nevertheless, seemingly no existing work has yet developed post-hoc explanations which can adapt to the case of full-fledged output text generation.

D Visualization of Explanations

We can gain insights into how our hierarchically structured interpretations give values at different levels, attributing importance to passages from different documents and then further localizing these attributions to the sentence and word level. We also provide an interactive version of the following visualizations hosted here.



#1	"Wilhelm Conrad Röntgen"
#2	"Marie Curie"
#3	"Ramón y Cajal"
#4	"Ernest Lawrence"

Output Classes: ○ 1 ○ 2 ○ 3 ● 4 ○ 5

Text Level: ○ paragraph ○ sentence ● word

Document [1] (Title: List of Nobel laureates in Physics) The first Nobel Prize in Physics was awarded in 1901 to Wilhelm Conrad Röntgen, of Germany, who received 150, 782 SEK, which is equal to 7,731,004 SEK in December 2007. John Bardeen is the only laureate to win the prize twice—in 1956 and 1972. Maria Sk<unk>odowska-Curie also won two Nobel Prizes, for physics in 1903 and chemistry in 1911. William Lawrence Bragg was, until October 2014, the youngest ever Nobel laureate; he won the prize in 1915 at the age of 25. Two women have won the prize: Curie and Maria Goeppert-Mayer (1963). As of 2017, the prize has been awarded

Document [2] (Title: Asian Americans in science and technology) Prize in physics for discovery of the subatomic particle J/<unk>. Subrahmanyan Chandrasekhar shared the 1983 Nobel Prize in Physics and had the Chandra X-ray Observatory named after him. Steven Chu shared the 1997 Nobel Prize in Physics for his research in cooling and trapping atoms using laser light. Daniel Tsui shared the 1998 Nobel Prize in Physics in 1998 for helping discover the fractional Quantum Hall effect. Yoichiro Nambu received the 2008 Nobel Prize in Physics for his work on the consequences of spontaneously broken symmetries in field theories. In 2009, Charles K. Kao was awarded Nobel Prize in Physics

Document [3] (Title: Scientist) and pursued through a unique method, was essentially in place. Ramón y Cajal won the Nobel Prize in 1906 for his remarkable observations in neuroanatomy. Marie Curie became the first female to win the Nobel Prize and the first person to win it twice. Her efforts led to the development of nuclear energy and Radio therapy for the treatment of cancer. In 1922, she was appointed a member of the International Commission on Intellectual Co-operation by the Council of the League of Nations. She campaigned for scientist's right to patent their discoveries and inventions. She also campaigned for free access

Document [4] (Title: Norwegian Americans) science, Ernest Lawrence won the Nobel Prize in Physics in 1939. Lars Onsager won the 1968 Nobel Prize in Chemistry. Norman Borlaug, father of the Green Revolution, won the Nobel Peace Prize in 1970. Christian B. Anfinsen won the Nobel Prize for chemistry in 1972. Ivar Giaever won the Nobel Prize in Physics 1973. Carl Richard Hagen is noted for his work in physics. In engineering, Clayton Jacobson II is credited with the invention of the modern personal watercraft. Ole Singstad was a pioneer of underwater tunnels. Ole Evinrude invented the first outboard motor with practical commercial application, recognizable today

Document [5] (Title: Maria Goeppert Mayer) of US postage stamps, along with Melvin Calvin, Asa Gray, and Severo Ochoa. Her papers are in the Geisel Library at the University of California, San Diego, and the university's physics department is housed in Mayer Hall, which is named after her and her husband. Maria Goeppert Mayer Maria Goeppert Mayer (June 28, 1906 – February 20, 1972) was a German-born American theoretical physicist, and Nobel laureate in Physics for proposing the nuclear shell model of the atomic nucleus. She was the second woman to win a Nobel Prize in physics, the first being Marie Curie. A graduate of the

Document [6] (Title: University of Chicago) Medal, which is rewarded annually to the best economist under the age of 40, has also been awarded to 4 current members of the university faculty. Notable faculty in physics have included the speed of light calculator A. A. Michelson, elementary charge calculator Robert A. Millikan, discoverer of the Compton Effect Arthur H. Compton, the creator of the first nuclear reactor Enrico Fermi, "the father of the hydrogen bomb" Edward Teller, "one of the most brilliant and productive experimental physicists of the twentieth century" Luis Walter Alvarez, Murray Gell-Mann who introduced the quark, second female Nobel laureate Maria Goeppert-Mayer, the

Document [7] (Title: École normale supérieure (Paris)) was also awarded the Abel prize. In addition, eight "normaliens" have gone on to receive the Nobel Prize in Physics: Claude Cohen-Tannoudji, Pierre-Gilles de Gennes, Albert Fert, Alfred Kastler, Gabriel Lippmann, Louis Néel, Jean Baptiste Perrin and Serge Haroche, while other ENS physicists include such major figures as Paul Langevin, famous for developing Langevin dynamics and the Langevin equation. Alumnus Paul Sabatier won the Nobel Prize in Chemistry. A ranking of universities worldwide based on ratios of alumni to Nobel prize-winners published in 2016 by American scholars Stephen Hsu and Jonathan Wai placed ENS as the first university worldwide, far

Document [8] (Title: Marie Curie) named in her honor. Marie Curie Marie Sk<unk>odowska Curie (; ; ; born Maria Salomea Sk<unk>odowska; 7 November 18674 July 1934) was a Polish and naturalized-French physicist and chemist who conducted pioneering research on radioactivity. She was the first woman to win a Nobel Prize, the first person and only woman to win twice, the only person to win a Nobel Prize in two different sciences, and was part of the Curie family legacy of five Nobel Prizes. She was also the first woman to become a professor at the University of Paris, and in 1995 became the first woman

Document [9] (Title: St John's College, Cambridge) Nobel Prize winners: Sir Edward Appleton, for discovering the Appleton layer, Sir John Cockcroft KCB, physicist who first split the atom, Allan Cormack, for the invention of the CAT scan, Paul Dirac, one of the founders of quantum mechanics, Sir Nevill Francis Mott, for work on the behaviour of electrons in magnetic solids, Abdus Salam, for unifying the electromagnetic force and the weak force, Frederick Sanger, molecular biologist, Maurice Wilkins, awarded Nobel prize for Medicine or Physiology with Watson and Crick for discovering the structure of DNA, and Eric Maskin (visiting & honorary fellow), awarded the Nobel Memorial Prize in

Document [10] (Title: Arthur H. Rosenfeld) Nobel Prize-winning Italian physicist. Rosenfeld coauthored a book on nuclear physics with Fermi, who was noted for building the world's first nuclear reactor. In 1954, after earning his PhD in physics, and with a recommendation from Fermi, Rosenfeld accepted a position as a teaching physicist at the University of California, Berkeley. At Berkeley, Rosenfeld joined the University of California Department of Physics and the particle physics research group at Lawrence Berkeley National Lab (LBNL) led by Nobel Laureate Luis Alvarez. Alvarez went on to win the Nobel Prize with research backed by his team of scientists that included Rosenfeld. In

Figure 9: Example explanation showing the different levels of the hierarchy. We see the correct answer of “Wilhelm Conrad Rontgen" highlighted in blue as the most important, and we can find the relevant words inside of the larger paragraph. The second most likely answer, Marie Curie, is highlighted within the 5th passage and we localize to the most relevant sentences.

E LLM Hallucinations

A problem of increasing importance is the issue of AI hallucinations created by LLMs (Dziri et al., 2022). With the wide dissemination of AI dialogue agents, there is a larger demand than ever before to resolve the longstanding problem of factual inaccuracy or hallucinations made by AI language models. In particular, the increased usage of LLMs as all-purpose information assistants has also added for the need to provide factual details to users requesting (and often expecting) accurate information.

Alongside larger service pipelines which will accurately branch to additional tools such as calculators, compilers, or external APIs, retrieval augmented generation is one of the leading candidate for ensuring the factuality of statements in text-generation provided by dialogue systems (Shuster et al., 2021; Huang et al., 2023b). Providing trusted source documents which are related to the current conversation topic or specifically answer a requested question is likely one of the only ways to continue to assert factual information across nearly the entire spectrum of human knowledge.

In accordance with the ultimate need for trusted sources of information to ensure the factuality of LLM-generated outputs, we envision a key application of TextGenSHAP will be towards eliminating hallucinations in LLMs. Existing methods have already demonstrated that the research-and-revise workflow is able to significantly reduce model hallucinations (Shuster et al., 2021; Gao et al., 2023). With the introduction of TextGenSHAP, we not only enable the improved retrieval step demonstrated in the main body of this work, but also further enable a cycle of improvement and human-in-the-loop feedback. Allowing human insight further enables previously successful methods like RLHF (Ouyang et al., 2022) to be incorporated to increase factuality and reduce hallucinations.

F Further Analysis for Dataset Repair on the MIRACL Dataset

In this section we dive into specific example queries and passages found from within the MIRACL dataset to analyze how appropriately they are being judged. For each example, we provide the question being asked and a table of relevant passages. In particular, for each query we provide the top-three rated passages according to the Shapley value computed for the query. In addition, we provide some of the most relevant passages which were not significantly considered by the Shapley value or those which were specifically rated by the MIRACL dataset (are one of the ten total passages which have a positive/relevant or negative/irrelevant label.) We cover three main types of examples to try to give a good coverage of which differences exist across the interpretations and across the dataset labels.

F.1 Erroneous Labels

These examples represent the relatively serious scenario where the original labels from the MIRACL dataset are found to be erroneous after exploration with our interpretabile explanations. We find that the selected passages from the explanation scores allow for us to quickly discover incorrect labels by finding the most important passages from a large corpus of potentially relevant information. In Table 7, we see that the original dataset mislabels paragraphs as irrelevant when they actually contain relevant information about grasshoppers' diets. In Table 8, we see that the human annotator actually mistakes the 'dialect test' with the 'dialectal method', causing incorrect labeling of the passages.

F.2 Insufficient Labels

These examples represent the relatively benign scenario where all labels are seemingly correct, but there is still an abundance of unlabeled passages which contain all of the necessary information. In particular, we highlight examples in Tables 9 and 10 where our method effectively locates passages which accurately answer the original query, but which are not in the top ten originally-retrieved passages from the information retrieval system. This paucity of label information in the MIRACL dataset restricts our method from its fullest potential when we consider the AUC metric only using the MIRACL's top ten labels. It is for this reason we consider utilizing the psuedolabel evaluation in the main text as a better signal for the end-to-end ODQA task.

F.3 Explanations Insufficient

In the final set of examples, we show the case where the explanations from the LLM identify incorrect passages. In Table 11, when looking for the origin of quantum field theory, the model focuses on the paper by Born, Heisenberg, and Jordan. Although extremely related, this work is generally considered a precursor to what is called quantum field theory rather than its first paper (Kuhlmann, 2023). In Table 12, we see the results finding the date of establishing the state flower of Texas. Although the highest rated explanation is a relevant passage, the next two highest have information both about Texan history and about the bluebonnet, but do not have the necessary dates to answer the question. We envision that even for such cases our method will still be useful for dataset construction and repair: since our method finds more relevant and more closely ambiguous paragraphs than existing retrieval-based systems, one will be able to more effectively utilize human annotators when using our method.

Shapley Ranking	MIRACL Rating	True Rating	Label Agreement	Title	Text
1st	Relevant	Relevant	Good	Grasshopper	Grasshoppers eat large quantities of foliage both as adults and during their development, and can be serious pests of arid land and prairies. Pasture, grain, forage, vegetable and other crops can be affected. Grasshoppers often bask in the sun, and thrive in warm sunny conditions, so drought stimulates an increase in grasshopper populations. A single season of drought is not normally sufficient to stimulate a major population increase, but several successive dry seasons can do so, especially if the intervening winters are mild so that large numbers of nymphs survive. Although sunny weather stimulates growth, there needs to be an adequate food supply for the increasing grasshopper population. This means that although precipitation is needed to stimulate plant growth, prolonged periods of cloudy weather will slow nymphal development.
2nd	Irrelevant	Relevant	Erroneous	Grasshopper	Grasshoppers are plant-eaters, with a few species at times becoming serious pests of cereals, vegetables and pasture, especially when they swarm in their millions as locusts and destroy crops over wide areas. They protect themselves from predators by camouflage; when detected, many species attempt to startle the predator with a brilliantly-coloured wing-flash while jumping and (if adult) launching themselves into the air, usually flying for only a short distance. Other species such as the rainbow grasshopper have warning coloration which deters predators. Grasshoppers are affected by parasites and various diseases, and many predatory creatures feed on both nymphs and adults. The eggs are the subject of attack by parasitoids and predators.
3rd	Irrelevant	Relevant	Erroneous	Grasshopper	Most grasshoppers are polyphagous, eating vegetation from multiple plant sources, but some are omnivorous and also eat animal tissue and animal faeces. In general their preference is for grasses, including many cereals grown as crops. The digestive system is typical of insects, with Malpighian tubules discharging into the midgut. Carbohydrates are digested mainly in the crop, while proteins are digested in the ceca of the midgut. Saliva is abundant but largely free of enzymes, helping to move food and Malpighian secretions along the gut. Some grasshoppers possess cellulase, which by softening plant cell walls makes plant cell contents accessible to other digestive enzymes.
–	Irrelevant	Irrelevant	Good	Kosher locust	In 1911, Abraham Isaac Kook, the chief rabbi of Ottoman Palestine, addressed a question to the rabbinic Court at Sanaá concerning their custom of eating grasshoppers, and whether this custom was observed by observing their outward features, or by simply relying upon an oral tradition. The reply given to him by the court was as follows: “The grasshoppers which are eaten by way of a tradition from our forefathers, which happen to be clean, are well-known unto us. But there are yet other species which have all the recognizable features of being clean, yet do we practice abstaining from them. [Appendage]: The clean grasshoppers () about which we have a tradition are actually three species having each one different coloration [from the other], and each of them are called by us in the Arabian tongue, “ğarād" (locusts). But there are yet other species, about which we have no tradition, and we will not eat them. One of which is a little larger in size than the grasshoppers, having the name of “’awsham". There is yet another variety, smaller in size than the grasshopper, and it is called “hanājir" (katydids).
–	Irrelevant	Irrelevant	Good	North American least shrew	Its diet consists of mostly small invertebrates, such as caterpillars, beetle larvae, earthworms, centipedes, slugs, and sow bugs. It will also eat from the corpses of dead animals, and small amounts of seeds or fruits. This shrew will eat its prey whole, but when eating crickets and grasshoppers, the North American least shrew will bite off the head of its prey and eat only the internal organs. When fighting a larger creature, it will aim for the legs and try to cripple its adversary, and will bite lizards, which are often too large for it to kill, on the tail, which then falls off and provides it with a meal while the lizard escapes. The North American least shrew will also sometimes live inside beehives and eat all the larvae. It will often share its food with other shrews. It eats more than its body weight each day and is known to store food.

Table 7: Examples from the MIRACL Dataset for the query of “What do Grasshoppers eat?”

Shapley Ranking	MIRACL Rating	True Rating	Label Agreement	Title	Text
1st	Unrated	Relevant	Okay	Interpersonal communication	A dialectical approach to interpersonal communication was developed by scholars Leslie Baxter and Barbara Montgomery. Their dialectical approach revolves around the notions of contradiction, change, praxis, and totality. Influenced by Hegel, Marx, and Bakhtin, the dialectical approach is informed by an epistemology that refers to a method of reasoning by which one searches for understanding through the tension of opposing arguments. Utilizing the dialectical approach, Baxter and Montgomery developed two types of dialectics that function in interpersonal relationships: internal and external. These include autonomy-connection, novelty-predictability, openness-closedness.
2nd	Unrated	Relevant	Okay	Dialectical research	Dialectical research or dialectical inquiry or dialectical investigation is a form of qualitative research which utilizes the method of dialectic, aiming to discover truth through examining and interrogating competing ideas, perspectives or arguments. Dialectical research can be seen as a form of exploratory research, in that there is not so much a research hypothesis to be tested, but rather new understandings to be developed.
3rd	Unrated	Relevant	Okay	Dialectic	Dialectic or dialectics (, “dialektike"; related to dialogue), also known as the dialectical method, is at base a discourse between two or more people holding different points of view about a subject but wishing to establish the truth through reasoned arguments. Dialectic resembles debate, but the concept excludes subjective elements such as emotional appeal and the modern pejorative sense of rhetoric. Dialectic may be contrasted with the didactic method, wherein one side of the conversation teaches the other. Dialectic is alternatively known as minor logic, as opposed to major logic or critique.
–	Relevant	Irrelevant	Erroneous	Dialect Test	The Dialect Test was created by A.J. Ellis in February 1879, and was used in the fieldwork for his work “On Early English Pronunciation". It stands as one of the earliest methods of identifying vowel sounds and features of speech. The aim was to capture the main vowel sounds of an individual dialect by listening to the reading of a short passage. All the categories of West Saxon words and vowels were included in the test so that comparisons could be made with the historic West Saxon speech as well as with various other dialects.
–	Irrelevant	Relevant	Erroneous	Frankfurt School	The Institute also attempted to reformulate dialectics as a concrete method. The use of such a dialectical method can be traced back to the philosophy of Hegel, who conceived dialectic as the tendency of a notion to pass over into its own negation as the result of conflict between its inherent contradictory aspects. In opposition to previous modes of thought, which viewed things in abstraction, each by itself and as though endowed with fixed properties, Hegelian dialectic has the ability to consider ideas according to their movement and change in time, as well as according to their interrelations and interactions.

Table 8: Examples from the MIRACL Dataset for the query of “When is the dialectical method used?”

Shapley Ranking	MIRACL Rating	True Rating	Label Agreement	Title	Text
1st	Relevant	Relevant	Good	List of songs in Guitar Hero Live	“Guitar Hero Live" is a 2015 music video game that’s developed by FreeStyleGames and published by Activision. It is the first title in the “Guitar Hero" series since it went on hiatus after 2011, and the first game in the series available for 8th generation video game consoles (PlayStation 4, Wii U, and Xbox One). The game was released worldwide on 20 October 2015 for these systems as well as the PlayStation 3, Xbox 360, and iOS devices including the Apple TV.
2nd	Unrated	Relevant	Okay	List of songs in Guitar Hero Live	Two hundred songs were initially available on GHTV on the game’s release on 20 October 2015.
3rd	Unrated	Relevant	Okay	Guitar Hero	Following a five-year hiatus, as described below, Activision announced “Guitar Hero Live" for release in late 2015 on most seventh-generation and eighth-generation consoles. “Live" was developed to rebuild the game from the ground up, and while the gameplay remains similar to the earlier titles, focusing primarily on the lead guitar, it uses a 3-button guitar controller with each button having “up" and “down" positions, making for more complex tabulators. The game using live footage of a rock concert, taken from the perspective of the lead guitarist, as to provide a more immersive experience.
–	Relevant	Relevant	Good	Guitar Hero	In 2015, Activision announced the first new title to the series in 5 years, “Guitar Hero Live", released in October 2015. The title is considered a reboot of the series, with development being performed by FreeStyleGames, who had developed the “DJ Hero" games previously. As of December 1, 2018, Activision disabled the GHTV servers for Guitar Hero Live, reducing playable content from approximately 500 songs to 42 on disc tracks.
–	Irrelevant	Irrelevant	Good	Guitar Hero Live	In an earnings report shortly following the gameś release, Activision stated that “Guitar Hero Live" was outselling their previous two “Guitar Hero" games, “" and “Guitar Hero 5", though did not report exact sales numbers. In their quarterly earnings results presented in February 2016, Activision reported that sales for “Guitar Hero Live" missed their expectations, and in March 2016, announced that they had to let go of about 50 of FreeStyleGamesémployees, though the studio still remains open to continue additional work for Activision. Prior to the Electronic Entertainment Expo 2016, Activision stated they will continue to produce content for “Guitar Hero Live" but have no present plans for another game.

Table 9: MIRACL Dataset Example for: “When was Guitar Hero Live first released?”

Shapley Ranking	MIRACL Rating	True Rating	Label Agreement	Title	Text
1st	Unrated	Relevant	Okay	Origin of Hangul	The Korean alphabet is the native script of Korea, created in the mid fifteenth century by King Sejong, as both a complement and an alternative to the logographic Sino-Korean “hanja". Initially denounced by the educated class as “eonmun" (vernacular writing), it only became the primary Korean script following independence from Japan in the mid-20th century.
2nd	Unrated	Relevant	Okay	Hangul	The Korean alphabet, known as Hangul (; from Korean ,), has been used to write the Korean language since its creation in the 15th century by King Sejong the Great. It may also be written following the standard Romanization.
3rd	Unrated	Relevant	Okay	Jeong In-ji	He is perhaps best known for having written the postscript of the “Hunmin Jeongeum Haerye", the commentary on and explanation of the native alphabet Hangeul invented by King Sejong in 1443. He also contributed to the “Goryeo-sa", the official history of Goryeo dynasty, and the “Yongbi Eocheon-ga".
–	Relevant	Relevant	Good	Korea	The Korean alphabet hangul was also invented during this time by King Sejong the Great.
–	Relevant	Relevant	Good	Origin of Hangul	Hangul was personally created and promulgated by the fourth king of the Joseon dynasty, Sejong the Great. Sejong’s scholarly institute, the Hall of Worthies, is often credited with the work, and at least one of its scholars was heavily involved in its creation, but it appears to have also been a personal project of Sejong.

Table 10: MIRACL Dataset Example for: “Who invented Hangul?”

Shapley Ranking	MIRACL Rating	True Rating	Label Agreement	Title	Text
1st	Irrelevant	Irrelevant	Good	Quantum field theory	Through the works of Born, Heisenberg, and Pascual Jordan in 1925-1926, a quantum theory of the free electromagnetic field (one with no interactions with matter) was developed via canonical quantization by treating the electromagnetic field as a set of quantum harmonic oscillators. With the exclusion of interactions, however, such a theory was yet incapable of making quantitative predictions about the real world.
2nd	Unrated	Irrelevant	Okay	History of quantum field theory	In 1925, Werner Heisenberg, Max Born, and Pascual Jordan constructed just such a theory by expressing the field's internal degrees of freedom as an infinite set of harmonic oscillators, and by then utilizing the canonical quantization procedure to these oscillators; their paper was published in 1926. This theory assumed that no electric charges or currents were present and today would be called a free field theory.
3rd	Unrated	Irrelevant	Okay	Quantum field theory	In 1913, Niels Bohr introduced the Bohr model of atomic structure, wherein electrons within atoms can only take on a series of discrete, rather than continuous, energies. This is another example of quantization. The Bohr model successfully explained the discrete nature of atomic spectral lines. In 1924, Louis de Broglie proposed the hypothesis of wave-particle duality, that microscopic particles exhibit both wave-like and particle-like properties under different circumstances. Uniting these scattered ideas, a coherent discipline, quantum mechanics, was formulated between 1925 and 1926, with important contributions from de Broglie, Werner Heisenberg, Max Born, Erwin Schrödinger, Paul Dirac, and Wolfgang Pauli.
–	Unrated	Relevant	Okay	History of quantum field theory	The first reasonably complete theory of quantum electrodynamics, which included both the electromagnetic field and electrically charged matter as quantum mechanical objects, was created by Paul Dirac in 1927. This quantum field theory could be used to model important processes such as the emission of a photon by an electron dropping into a quantum state of lower energy, a process in which the "number of particles changes"—one atom in the initial state becomes an atom plus a photon in the final state. It is now understood that the ability to describe such processes is one of the most important features of quantum field theory.
–	Relevant	Relevant	Good	History of quantum field theory	The third thread in the development of quantum field theory was the need to handle the statistics of many-particle systems consistently and with ease. In 1927, Pascual Jordan tried to extend the canonical quantization of fields to the many-body wave functions of identical particles using a formalism which is known as statistical transformation theory; this procedure is now sometimes called second quantization. In 1928, Jordan and Eugene Wigner found that the quantum field describing electrons, or other fermions, had to be expanded using anti-commuting creation and annihilation operators due to the Pauli exclusion principle (see Jordan–Wigner transformation). This thread of development was incorporated into many-body theory and strongly influenced condensed matter physics and nuclear physics.

Table 11: MIRACL Dataset Example for: "When was quantum field theory developed?"

Shapley Ranking	MIRACL Rating	True Rating	Label Agreement	Title	Text
1st	Relevant	Relevant	Good	Bluebonnet (plant)	Bluebonnet is a name given to any number of blue-flowered species of the genus “Lupinus" predominantly found in southwestern United States and is collectively the state flower of Texas. The shape of the petals on the flower resembles the bonnet worn by pioneer women to shield them from the sun. Species often called bluebonnets include:On March 7, 1901, “Lupinus subcarnosus" became the only species of bluebonnet recognized as the state flower of Texas; however, “Lupinus texensis" emerged as the favorite of most Texans. So, in 1971, the Texas Legislature made any similar species of “Lupinus" that could be found in Texas the state flower.
2nd	Unrated	Irrelevant	Okay	John Nance Garner	Garner was elected to the Texas House of Representatives in 1898, and re-elected in 1900. During his service, the legislature selected a state flower for Texas. Garner fervently supported the prickly pear cactus for the honor, and thus earned the nickname “Cactus Jack". (The Bluebonnet was chosen.) In 1901 Garner voted for the poll tax, a measure passed by the Democratic-dominated legislature to make voter registration more difficult and reduce the number of black, minority, and poor white voters on the voting rolls. This disfranchised most minority voters until the 1960s, and ended challenges to Democratic power; Texas became in effect a one-party state.
3rd	Irrelevant	Irrelevant	Good	Alamo Fire	Maroon and white bluebonnets were developed as part of an effort to compose a Texas flag with red, white, and blue bluebonnets to celebrate Texas’ sesquicentennial in 1986. Pink bluebonnets were found in San Antonio, and reddish examples were selectively bred by Dr. Jerry Parsons of the Texas A&M AgriLife Extension Service to eventually give maroon bluebonnets in 2000. The color of these bluebonnets was fitting, as the color maroon is strongly associated with Texas A&M University.
–	Irrelevant	Irrelevant	Good	Bluebonnet Ordnance Plant	The plant was operated by the National Gypsum Company but overseen by the military and was one of the four Ordnance plants in the United States during World War II. The army engineers were in charge of all plant construction while the Gypsum personnel and others worked out other strategies. Bluebonnet Ordnance Plant got its name from Major Paul Van Tuyl, who named the plant after the state flower of Texas (Bluebonnet).
–	Irrelevant	Irrelevant	Good	Lupinus texensis	Lupinus texensis, the Texas bluebonnet or Texas lupine is a species of lupine endemic to Texas. With other related species of lupines also called bluebonnets, it is the state flower of Texas.

Table 12: MIRACL Dataset Example for: “When were bluebonnets named the state flower of Texas?”