Exploration of Closed-Domain Question Answering Explainability Methods With a Sentence-Level Rationale Dataset

Lize Pirenne*, **Samy Mokeddem***, **Damien Ernst**, **Gilles Louppe**

*Equal contributions, Université de Liège,
lize.pirenne@uliege, samy.mokeddem@uliege, dernst@uliege, g.louppe@uliege

Abstract

In this paper, we address the problem of Rationale Extraction (RE) from Natural Language Processing: given a context (C), a related question (Q) and its answer (A), the task is to find the best sentence-level rationale (R^*). This rationale is loosely defined as being the subset of sentences of the context C such that producing A would require at least R^*. We have constructed a database where each entry is composed of the four terms (C, Q, A, R^*) to explore different methods in the particular case where the answer is one or multiple full sentences. The methods studied are based on TF-IDF scores, embedding similarity, classifiers and attention and have been evaluated using a sentence overlap metric akin to the Intersection over Union (IoU). Results show that the best scores were achieved by the classifier-based approach. Additionally, we observe the growing difficulty of finding R as the number of sentences in the context increased. Finally, we underlined a correlation in the case of the attention-based method between its performance and the ability of the underlying large language model to provide given C and Q an answer similar to A.

1 Introduction

Reliable Question and Answer (QA) systems are as useful as they are challenging to implement. Even in the Closed-Domain Question Answering (CQA) task, where the answer is restricted by the information explicitly provided within the context, hallucinations can be interleaved in or substitute the answer sought.

The setting of CQA appears regularly in modern QA systems thanks to advances in Retrieval Augmented Generation (RAG) (Lewis et al. 2020b). The retrieved documents are considered factually correct and answering the question becomes only a matter of extracting information from them. This is often the case for customer service chatbots or enterprise-wide dynamic knowledge bases. In both cases, avoiding hallucinations and ensuring that the answer is grounded in reality is a priority.

Inside the context can lie both relevant and irrelevant information to the question asked. Using the hypothesis that there is no redundant statement inside the context, we can identify the smallest set of sentences in the context that is required for producing the answer to the question, which we call the sentence-level rationale. For conciseness, we will refer to it as the rationale.

Extracting the rationale of an answer A from a given context C and a question Q offers significant benefits for CQA systems (Sun et al. 2022). Indeed, they enhance explainability: by identifying the rationale behind an answer, users can gain insights into the decision-making process of the underlying model of the system and assess its reliability. This is particularly valuable in domains demanding high levels of trust and transparency, such as healthcare (Ribeiro, Singh, and Guestrin 2016) or legal applications (Chalkidis et al. 2021). Furthermore, finding the rationale can potentially improve the quality of generated responses. For example, research suggests that leveraging the rationale during prompt engineering can lead to better generation outcomes (Krishna et al. 2023). Others implicitly compare their generation against the rationale to lead the sampling away from hallucinations (Chuang et al. 2024).

Wiegreffe and Marasović (2021); Liu et al. (2024) provide an overview of existing datasets for rationale extraction, although many are for classification only. We identified Hotpot-QA (Yang et al. 2018) as a close match to our needs, but its main focus is to challenge models on multi-hop reasoning. Since we mainly want to assess the capacity of different methods to find explicit rationales, we have decided to annotate an existing CQA dataset.

We provide the following contributions:

- We bring additional annotations to a dataset such that it becomes tailored for sentence-level rationale extraction in closed-domain question answering with full-sentence answers.
- We investigate various methods for sentence-level rationale extraction and compare their performance on our dataset. We have explored attention-based, classifier-based, embedding similarity and TF-IDF methods.
- We study the effect of increasing the number of sentences in the context on performance and compare selection characteristics of the methods such as whether they use a threshold or a ranking approach.

The importance of the last point can be motivated by previous studies on large language models (LLMs) that have shown repeated weaknesses with increasing context size (now reaching more than a million tokens (Reid et al. 2024;

*These authors contributed equally.

Liu, Zaharia, and Abbeel 2023)); more tokens in the prompt seems to be inversely correlated with answer quality (Shi et al. 2023). Consequently, we have explored various methods and models, assessed their ability to find a rationale as the number of sentences in the context increased and discussed how scalable their rationale extraction mechanism is.

Related work

This section discusses how our paper relates to topics such as explainability, natural language processing and explanation regularisation and also discusses datasets for rationale extraction.

Explainability. Zhao et al. (2024) provide an in-depth survey of methods to enhance the explainability of LLMs. Our work aligns with the category of local explanation models defined in this survey. Local explanation models focus on explaining the output of a model based on its specific inputs, in contrast to global explanation models, which identify general patterns in its input data to explain phenomena such as accuracy degradation.

The majority of the methods explored here (all except the generation mode of the attention-based methods) can also be classified as attribution-based explanations using surrogate models. Attribution-based methods identify what importance to put to each input feature similar to SHAP (Lundberg and Lee 2017), Integrated Gradients (Sundararajan, Taly, and Yan 2017) or SmoothGrad (Smilkov et al. 2017). In our case, the importance is binary in nature: either a sentence (feature) is to be considered as part of the rationale or it is not. The term "surrogate model" refers to the fact that the model used for generating explanations is not the same as the model that produced the original output as is the case with the LIME framework (Ribeiro, Singh, and Guestrin 2016). Surrogate model-involving methods are also known as post hoc explanation methods, as discussed in AMPLIFY (Krishna et al. 2023).

Rationale Extraction in Natural Language Processing. The extraction of rationale from model inputs has been explored at different levels of granularity, such as token level (Moradi, Kambhatla, and Sarkar 2021; Yu et al. 2021) or sentence level (Glockner, Habernal, and Gurevych 2020). As in Moradi, Kambhatla, and Sarkar (2021), our attention-based method uses attention to extract the rationale, although they used attention in the supervised task of machine translation as a regularisation parameter. Moreover, as in Glockner, Habernal, and Gurevych (2020) some of the methods proposed in our work aim to extract the k most relevant sentences from the context based on a relevance measurement while others use a more traditional threshold. Lamm et al. (2021) calls this rationale extraction task explanation prediction.

GopherCite (Menick et al. 2022) produces the rationale in line by adding special tokens and learning to produce exact quotes between them. This technique allows for restricting the sampling process to only produce sentences that exist in the context, thereby ensuring the exactitude of the quote.

Various other frameworks, such as MARTA (Arous et al. 2021), and Ross, Hughes, and Doshi-Velez (2017), have proposed methods to enhance the explainability of machine learning models through rationale extraction. However, these frameworks are focused mostly on classification tasks, with only a few (Krishna et al. 2023; Chan et al. 2022) specifically addressing whole sentences as answers.

Explanation Regularisation. Explanation Regularisation (ER) (Joshi et al. 2022) explores how rationale can be used to provide supplementary training objectives for models. This can involve techniques such as introducing loss penalties that encourage the model to focus on informative parts of the context (Ross, Hughes, and Doshi-Velez 2017) or enforcing attention sparsity to prevent the model from becoming overwhelmed with excessive information (Moradi, Kambhatla, and Sarkar 2021). Frameworks like UNIREX (Chan et al. 2022) demonstrate how these methods that leverage rationales can be integrated into a larger system for improved CQA performance. Similarly, in our reinforcement learning attention-based method, we have regularised the reward by adding our explanatory metric.

Datasets for rationale extraction. There exists a number of datasets specialised in providing rationales. Excluding datasets that are limited to classification (like MultiRC (Khashabi et al. 2018), FEVER (Thorne et al. 2018) or Rationales-Movies (Zaidan, Eisner, and Piatko 2008)) and to the best of our knowledge, we have found nine relevant datasets. There is QED (Lamm et al. 2021) that has the strong assumption of there being only one sentence for the rationale which is rarely the case in our own examples. Then, there are AdversarialQA (Bartolo et al. 2020), Natural Questions (Kwiatkowski et al. 2019), MLQA (Lewis et al. 2020a), SQuAD (Rajpurkar et al. 2016; Rajpurkar, Jia, and Liang 2018) and TiDiQA (Clark et al. 2020) that have their answer directly extracted from the input, rendering the task too easy (a simple search). There are also QuoRef (Dasigi et al. 2019) and QuAC (Choi et al. 2018) which are more focused on solving co-references. Finally, Hotpot-QA (Yang et al. 2018) is good for our task but is quite challenging for smaller models due to the objective of using multi-hop reasoning.

2 Problem statement

Given the triplet question-context-answer (Q, C, A), we are interested in finding a method that uses this triplet to produce a good approximation R of the best rationale $R^* \in C$ to explain A. More formally, let $\mathcal{M}$ the set of all methods taking (Q, C, A) as input and outputting a subset of sentences R in the context C ($R \in C$). The objective is to find the method $M \in \mathcal{M}$ that provides a good approximation $R = M(Q, C, A)$ of R^*.

To identify a high-performing method $M \in \mathcal{M}$, we have at our disposal a training set $TS = \{(Q_i, C_i, A_i, R_i^*)\}_{i=1}^{N}$ where each i is composed of the (i) question, (ii) context, (iii) answer and (iv) rationale.

Moreover, given R^*, the quality of the approximated rationale R will be assessed using the Intersection-over-Union (IoU) score defined by

$$IoU(R,\ R^*) = \frac{|R \cap R^*|}{|R \cup R^*|}, \tag{1}$$

where the operator $|\ |$ gives the number of character in all sentences in the set it operates on, $\cup$ outputs the set of sentences that appear in at least one operand, and $\cap$ computes the set of sentences appearing in both operands. We note that this IoU is equivalent to $\frac{1}{\frac{1}{\text{Precision}}+\frac{1}{\text{Recall}}-1}$ if both precision and recall are also defined on a per character basis. This choice was motivated by its use in DeYoung et al. (2020) but we differ in that we work on characters rather than tokens and our R and R^* always correspond to complete sentences. We prefer working on sentences because we believe they are more interpretable for end-users and are easier to annotate.

The IoU score will be used in the training (reward regularisation), validation and evaluation sets to improve and assess the performance of a method M.

3 Methods

In this section, we will explain the four different methods, named Embedding Similarity, TF-IDF, LLM classifier and LLM attention, that will be later used in the experiments.

Embedding similarity

The first method tested is a sentence-embedding method based on LLMs pre-trained for Semantic Textual Similarity, which aims to determine the degree of similarity between two pieces of text.

To generate the embedding of a sentence using an LLM, the most commonly employed approaches are to either average the final hidden vectors (before the classification layer of a classical causal Transformer LLM) of the tokens in the sentence, or simply pool the final hidden vector of the special first token (the [CLS] token). We have chosen to use the latter.

We have defined two methods using two different cut-off functions: the first chooses the top k sentences with the highest scores and the other picks all sentences above a certain threshold t. They can more succinctly be presented as in Equations 2 and 3, where Embed() is a function that takes a sentence as input and returns a vector $v \in \mathbb{R}^{d_{embed}}$, with d_{embed} being the size of the embedding and cos_sim designating the cosine similarity function.

$$\begin{aligned}&\textbf{Embedder}_{\text{Top-k}}(A,\ C)\\&= \underset{s_j \text{ in } C}{\text{Top-k}}(\text{cos\_sim}(\text{Embed}(A), \text{Embed}(s_j)))\end{aligned} \tag{2}$$

$$\begin{aligned}&\textbf{Embedder}_{\text{Threshold}}(A,\ C)\\&= \left\{ s \in C \;\middle|\; \text{cos\_sim}\Big(\text{Embed}(A), \text{Embed}(s)\Big) > t \right\}\end{aligned} \tag{3}$$

TF-IDF

The second class of methods tested uses Term Frequency - Inverse Document Frequency (TF-IDF) (Salton and Buckley 1988) rather than an LLM to produce embeddings but otherwise operates the same as the previous method. Each column of the TF-IDF matrix corresponds to a term in the vocabulary and each row is a document. The value represents the TF-IDF score of the corresponding term in the document. This score, derived from the Term Frequency (TF) and Inverse Document Frequency (IDF) values, highlights terms that are prevalent within a document but rare across the corpus, thereby underlining their significance within that document.

The construction of the TF-IDF score is described by Algorithm 1. The brackets in the algorithm refers to the Iverson brackets, they produce "1" if the inside is true, and "0" otherwise. The function split_terms extracts each term composing its argument.

Algorithm 1 TF-IDF Fit and Transform

Inputs:
D : The corpus composed of n documents D_i

Outputs:
TF-IDF : The embedding matrix of the corpus

1: $V = \{\text{split\_terms}(D)\}$
2: $\text{TF}(t, d) = \frac{1}{|S|} \sum_{w \in S} [w = V_t], S = \text{split\_terms}(D_d)$
3: $\text{IDF}(t) = \log\left(\frac{1}{|D|} \sum_{d=0}^{|D|} [\text{TF}(t, d) > 0]\right) + 1$
4: $\text{TF-IDF}(t, d) = \text{TF}(t, d) * \text{IDF}(t)$

As before, we have tried both a threshold and a ranking approach, described in Equations 4 and 5, where TF-IDF() is a function that takes a sentence as input and returns a vector $v \in \mathbb{R}^{d_{voc}}$, with d_{voc} being the size of the vocabulary.

$$\begin{aligned}&\textbf{NG}_{\text{Top-k}}(A,\ C)\\&= \underset{s \text{ in } C}{\text{Top-k}}(\text{cos\_sim}(\text{TF-IDF}(A), \text{TF-IDF}(s_j)))\end{aligned} \tag{4}$$

$$\begin{aligned}&\textbf{NG}_{\text{Threshold}}(A,\ C)\\&= \left\{ s \in C \;\middle|\; \text{cos\_sim}\Big(\text{TF-IDF}(A), \text{TF-IDF}(s)\Big) > t \right\}\end{aligned} \tag{5}$$

LLM classifier

The third class of method that will be used in our experiments is inspired by (Sun et al. 2023) and (Chae and Davidson 2023). It involves fine-tuning a pre-trained LLM for binary text classification. The objective is to determine whether a sentence in the context is part of the rationale or not.

The input of the classifier consists in the concatenation of the sentence to be classified (w) surrounded by its first left and right neighbouring sentences, the answer text (A), and the question text (Q). This method can be formalised as:

$$\begin{aligned}&\textbf{LLM}_{\text{Classifier}}(Q,\ C,\ A)\\&= \left\{ s \in C \;\middle|\; \text{classify}((s,\ A,\ Q))^p > 0.5 \right\}\end{aligned} \tag{6}$$

Context:
This small mansion has medieval origins and is surrounded by a large landscaped park.<!>The present building was constructed in 1634 by Evan Edwards, a member of a well established Flintshire family which traced its descent from the Welsh king Hywel Dda.<!> He most likely incorporated an older medieval house into the north wing of the current building.
Answer:
Rhual was constructed in 1634 by Evan Edwards
Question:
When was Rhual constructed?

(a) Positive example

Context:
Rhual is a Grade I listed building in Flintshire. <!>This small mansion has medieval origins and is surrounded by a large landscaped park.<!> The present building was constructed in 1634 by Evan Edwards, a member of a well established Flintshire family which traced its descent from the Welsh king Hywel Dda.
Answer:
Rhual was constructed in 1634 by Evan Edwards
Question:
When was Rhual constructed?

(b) Negative example

Figure 1: Examples of the formatted input feed to the LLM classifier. The sentence to classify is highlighted within the context window.

where p denotes the positive label of the soft-maxed output of classify$((s,\ A,\ Q))$, representing the LLM classifier.

We provide an illustration of the classification procedure in Figure 1.

LLM attention

This last method is based upon the attention mechanism present in most LLMs. The attention relates two parts of the input together with a numerical value, akin to a correlation matrix. Incidentally, it is of the form $N \times N$, where N is the number of parts in the input (nicely explained in Cho et al. (2024)). These parts are called tokens.

The attention mechanism is replicated multiple times in a single layer, all with different weights (multi-head attention). This means that for a given model and for each token of the answer, there are $n_{layer} \times n_{head}$ attention results to consider, each attending to different parts of the input and enabling it to understand different linguistic features (Clark et al. 2019).

Our goal with this method is to produce a view of this matrix where we only consider how the context is related to the answer. Therefore we produce an aggregation over the tokens of the answer. An example of these aggregated (mean) values is shown in Figure 2 where the grey intensity is the projection of the obtained values onto RGB space where all channels are of equal values.

In essence, we transform the matrix presented in Figure 3 into the compression of the light-grey components along the ordinates. Then we map the tokens and strings together to be able to average over sentences.

We start from the internal values of attention $a(i,j)$ per token of the answer $i \in T(A)$ and of the context $j \in T(C)$, where $T()$ is the tokenizer. We average these values over the tokens of the answer to have only one per token of the context by following the equation: $\mathtt{A}(j) = \frac{1}{|T(A)|} \sum_{i \in T(A)} a(i,j)$.

We get the following criterion:

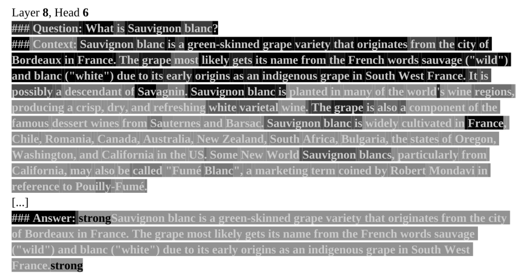

Figure 2: Average attention weights over a generation by Google/gemma-2b, colourised (darker is higher).

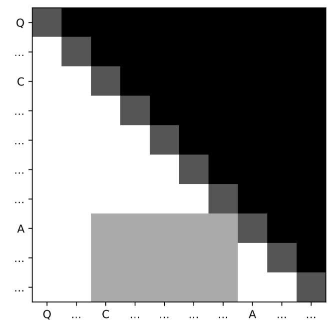

Figure 3: Representation of the attention matrix for each token of the question Q, context C and answer A. Black is the causal mask, dark grey is the predicted token and light grey represents the tokens the attention will be averaged on (along the axis of ordinates then by parts on the abscissas).

$$\textbf{LLM}_{Attention}(Q, C) = \underset{s \text{ in } C}{\text{Top-K}} \left(\left\{ s \;\middle|\; \left(\frac{1}{|s|} \sum_{j \in T(s)} \mathtt{A}(j) \right) > t \right\} \right) \tag{7}$$

.

4 Experiments

In this section we will describe the dataset that was used to evaluate our methods and how they were concretely implemented.

Dataset

The following paragraphs will elaborate on the construction of the reference dataset from which the training, validation and evaluation sets were extracted.

Data source and filtering. We specifically chose the closed-QA part of the `databricks-dolly-15k` (Conover et al. 2023) as our base CQA dataset. We filtered the triplets (Q,C,A) in the dataset by excluding those where the answer A did not respond to the question Q strictly using the context C. When little change was required to avoid discards (e.g., deleting a sentence, adding a word,...), we applied those instead. This filtered dataset contains 1595 triplets.

Construction. From the filtered CQA dataset, each triplet has undergone human annotation to form our Rationale Databricks Dolly CQA (RDD) dataset. The annotation process involves linking each (Q, A) pair to the relevant sentences within the context C. These form the rationale and will be denoted as R^*. We have labelled in the context only complete sentences rather than segments of sentences. In cases where multiple questions existed within the same example, each sub-question has been labelled separately. The annotation process was done in this manner so that a more complex problem statement could be created: in this new problem, the goal is to produce multiple sub-rationales corresponding to multiple sub-questions and answers; there is often a combination of questions in a single Q, such as inquiries for *what*, *who* and *when*, and the current statement ignores all these subdivisions. For the rest of this paper, we will consider that a data point i is represented as a quadruplet (Q_i, C_i, A_i, R_i) where the input x is the triplet (Q_i, C_i, A_i) and the targeted output y is the union of all sub-rationales R_i^n: $R_i = \bigcup_{n=1}^{N} R_i^n$. The tool we used to annotate the dataset is `Doccano` (Nakayama et al. 2018).

Dataset utilisation. The RDD dataset has been shuffled using the same random seed for all experiments to ensure consistency, and then divided into three sets: the training, validation and evaluation set. They represent respectively 80%, 10% and 10% of the original dataset. The training set has been used to train the methods, the validation set to finetune the parameters, and lastly the evaluation to compare their performance.

We decided to split the data points into four categories based on the number of sentences in the context as shown in Figure 4. To do so, we created four intervals $[1; 3]$, $[4; 6]$, $[7; 10]$ and $[11; \inf]$ of different sizes to keep the number of samples in each one comparable. In particular, the last category covers all triplets above ten sentences that have less than 2048 tokens for Q and C; its largest member has 75 sentences in the context. The token restriction removes two triplets that had 88 and 127 sentences.

Training

In this paragraph, we will discuss the different choices that have been made to run each experiments.

All methods explored were able to run on our two Nvidia 2080tis and will be further explained in this section. To achieve this, for the methods necessitating training, we have used Low Rank Adaptation (LoRA) (Hu et al. 2022) together with quantisation (called QLoRA (Dettmers et al. 2023)). LoRA is a technique to reduce the number of trainable parameters and quantisation reduces the representation space of the parameters to fit on a smaller number of bytes.

Embedding. For the experiments based on the embeddings, we utilised two pre-trained LLMs: `Sentence-Bert` (Reimers and Gurevych 2019), one of the pioneering models for text similarity embedding based on LLM, and `SFR-Embedding-Mistral`

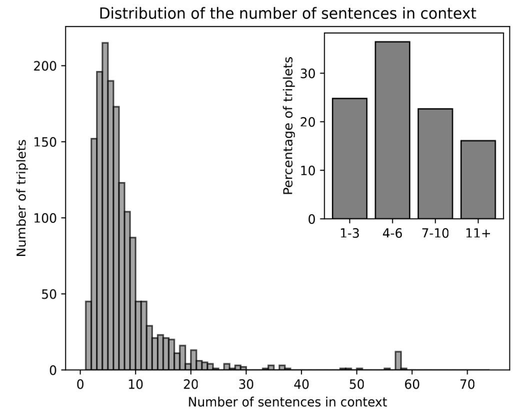

Figure 4: Truncated (max. 2048 tokens for Q and C) triplet distribution by number of sentences in the context.

(Meng et al. 2024), the current state-of-the-art (SOTA) of open-source models for textual similarity tasks according to (Muennighoff et al. 2023).

To choose the appropriate hyper-parameters, we have swept over $k \in [1 \ldots 5]$ and $t \in [0.1 \ldots 0.9]$ (90 steps).

TF-IDF. Since this method does not centre on a model, we did not have to make any specific choice other than the hyper-parameters, for which we have made the same sweep as for the embedding method.

LLM Classifier. The different pre-trained LLMs used to train classifiers are: `DistilBERT` (Sanh et al. 2020), `RoBERTa-Base` (Liu et al. 2019), and `Gemma-2B` (Gemma et al. 2024).

We have fine-tuned the different pre-trained models over 20 epochs using the standard cross-entropy loss. The selection of the best models and checkpoint is based on accuracy since the IoU metric does not apply to the input of the classifier; our metric is only used during evaluation.

The hyper-parameter selection is primarily based on empirical results, the final parameters used for all fine-tuned models are the following: Learning rate=5e-5, LoRa rank=4, alpha LoRa=4, LoRa dropout=0.1.

LLM Attention. For the class of methods using attention, we only used `Gemma-2B` (Gemma et al. 2024).

There are three variations of the attention-based method. The first (suffixed *Base* in tables and graphs) is the base pre-trained model. Consequently, by the definition of this method, this variation only has access to the pair (Q, C). The second (*FT*) is continually pre-trained (via Huggingface:Trainer) on the base CQA dataset by performing a standard causal language modelling training with Q, C and A always present concurrently. In this variation, the model leverages the triplet (Q, C, A). The third (*RL*) is an RL-tuned (via HuggingFace:TRL:PPOTrainer) version of the second, where the reward is the average of the IoU score and the METEOR (Banerjee and Lavie 2005) metric with flat penalty for not including the *EOS* token. This last variation has access to all the possible information during training. At testing time, we use the entire triplet (Q, C, A) to produce the rationale (we note that we can also generate the answer then find the rationale, but due to its lower quality, it gets 15% less IoU).

Due to hardware constraints, we have limited the sizes of the examples to $|(Q, C)| < 2048$ and $|A| < 542$ such that the total number of tokens respected $|(Q, C, A)| < 2600$. For the RL training, these values are respectively 450, 50 and 500.

The pre-training of the initial model is continued on the base dataset, ignoring the rationale. The prompts have been formatted by adding "### Question: ", "### Context: ", "### Answer: " and "### End" separated by double line breaks in order to provide a clear description of the task. The training was continued for one epoch.

The attention computation of Equation 7 requires the *head*, *layer*, *threshold* and k to be set. To do so, we have swept over all eight heads, 18 layers for $k \in [0 \ldots 4]$ (0 indicates no restriction) and $t \in [0.006, \ldots, 0.001, 0.0005, 0.0003, \ldots, 0.0]$ to compute the average score on the validation set and took the best combination of parameters. The range for t was motivated by an analysis of reoccurring values.

Results

In this section we will show and comment on the results given by our simulations. These have been obtained by running the best model we obtained on the evaluation set once.

Best method. As can be seen in Table 1 summarising the performance of all methods over the entire test set, the best method is the Gemma classifier, which shows a small improvement over the RoBERTa classifier. However, when taking into consideration only the largest context size in Figure 5, the attention method seems to have a slight edge over the classifier.

Influence of model size. For classifiers, Table 1 suggest that larger models performs better at our small scale, or that models able to generate such answers are also more likely to find the correct rationale. This is also the case for embedding-based methods, as SFR-Embedding-Mistral consistently gets a higher IoU score than Sentence-Bert. It would be interesting to see the same comparison for the attention method, which could also fix the consistent drop in IoU score we observed while using the model to generate the answer.

Influence of hyper-parameter. The sweep of hyper-parameters has shown that methods using a ranking approach perform best with small k values (i.e., 1 or 2), except the attention method which does not seem to have the same flaw. This is likely due to the skew of the dataset for smaller numbers of sentences, as shown in Figure 4. For methods using Top-k in particular, we can observe that they are capped around a 0.7 IoU score in the first group (1 to 3), likely because $k = 1$ restricts them from retrieving additional sentences needed for a higher score. In contrast, threshold methods do not have this limitation and can theoretically achieve an IoU score of 1 (i.e., the maximum score). We note that while the head and layer change in the reported table for the fine-tuned attention method, our sweep showed only a 0.1% difference on the IoU score.

Model	Size	IoU
Sentence-Bert-large (k=1)	109M	$0.61_{\pm 0.05}$
Sentence-Bert-large (t=0.68)	109M	$0.54_{\pm 0.06}$
SFR-Embedding-Mistral (k=1)	7.11B	$0.65_{\pm 0.05}$
SFR-Embedding-Mistral (t=0.72)	7.11B	$0.59_{\pm 0.06}$
TF-IDF (k=1)	/	$0.64_{\pm 0.05}$
TF-IDF (t=0.25)	/	$0.66_{\pm 0.05}$
Classifier DistilBERT	67M	$0.64_{\pm 0.06}$
Classifier RoBERTa	125M	$0.75_{\pm 0.05}$
Classifier Gemma	2.51B	$\mathbf{0.79_{\pm 0.04}}$
Gemma Base (L=5, H=4, k=0, t=0.002)	2.51B	$0.74_{\pm 0.05}$
Gemma FT (L=8, H=6, k=0, t=0.002)	2.51B	$0.75_{\pm 0.05}$
Gemma RL (L=5, H=4, k=0, t=0.002)	2.51B	$0.76_{\pm 0.05}$

Table 1: Summary table of experiment results, including the number of parameters for each method (size) and the average IoU score obtained on the evaluation set, presented with a 95% confidence interval assuming a student-t distribution.

Influence of training on attention. The results of Table 1 demonstrate that the attention method performs marginally better on longer contexts after training the LLM model on the training dataset despite not being trained for this metric in particular, highlighting a possible correlation between answer generation capability and rationale extraction. In our setting, RL training does not significantly improve the results on the evaluation or test set despite showing a 5% absolute increase at training time. This can indicate that the SFT training was sufficient and going beyond would only over-fit.

Influence of the number of sentences in the context. As shown in Figure 5, for all methods, as the number of sentences in the context increases, the performance of the models decrease. This supports the results of Atanasova et al. (2022) who reported (converted from Precision and Recall) over three datasets of increasing size IoUs of 0.89, 0.66 and 0.59 (see Appendix B). The projected, context size weighted, scores of our classifier would be 0.83, 0.64 and 0.62. These similar scores show that this trend is generalised beyond our dataset.

Limitations

In this section we will discuss some limitations we encountered and/or are aware of.

Concerning the dataset, it has only been annotated by us and thus may not have been reviewed impartially and/or in sufficient depth. Additionally, we could have extended the dataset by using existing datasets for rationale extraction in classification and procedurally generated appropriate outputs. However, this approach would decrease the variety of answers and the impact on performance of such a decrease has, to our knowledge, not yet been studied. We could

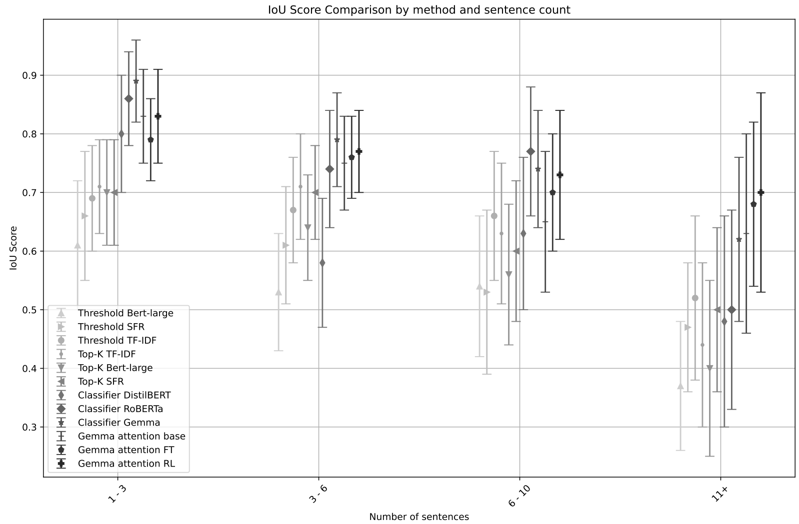

Figure 5: IoU scores with 95% confidence error bars (student-t).

also have used HotpotQA (Yang et al. 2018) to broaden the dataset.

Despite our best attempt at exploring a widely applicable array of methods, they still come with intrinsic restrictions. For example, the attention-based method cannot be extended easily to sub-quadratic (Kitaev, Kaiser, and Levskaya 2020; Ding et al. 2023; Ma et al. 2023; Choromanski et al. 2021; Song et al. 2023), or attention-less (Gu and Dao 2023; Zhai et al. 2021; De et al. 2024; Peng et al. 2023; Beck et al. 2024) LLMs. The exploration of gradient-based methods would have been more broadly applicable but would require many additional backward passes inducing a higher compute requirement. This particular choice has been at the center of debates as discussed in (Bastings and Filippova 2020).

The methods studied also do not cover all existing ones and should not be regarded as an exhaustive comparison but rather an educated guess of potentially well-performing methods.

The models used also were limited by our hardware, being two Nvidia 2080ti. Some runs were carried out on a cluster to hasten the experiments but the parameters have been kept the same such that any of these can be exactly replicated on the original hardware requirements.

Finally, the tuning of hyper-parameters required a lot of compute, so we have restricted the range of values to those we estimated which would be most pertinent. The linear/grid search approach could also be revised to other optimisation techniques.

5 Conclusion

In this paper, we have constructed the CQA Rationale Databricks Dolly dataset for the express purpose of improving sentence-level rationale extraction in closed-domain question answering where the answer appears as full sentences. This dataset will be provided under the same license (CC BY-SA 3.0) as the original for the community along with all the code used for the experiments. We have studied a range of methods using the dataset to foster interest in the subject that have been evaluated via our newly introduced IoU metric for sentences.

From our results, we have underlined the difficulty in reaching satisfying results as the scale of the context grows. Nonetheless, we have found that classifier models could achieve an IoU of **79%**, which is on a par with previous work for smaller sizes of contexts (a few sentences). Moreover, the attention method shows a more robust trend as the size of the context increase. Still, achieving satisfying results on lengthy documents remains an unresolved challenge.

This research calls for several future works. First, in the produced dataset there have been numerous instances of questions that contained sub-questions while taking care of differentiating the sub-answers and corresponding sub-rationales. This is of particular interest because no other dataset reviewed seemed to differentiate this case.

Second, it may be interesting to see other approaches compete, such as gradient-based methods, to see how the scores could be improved. This includes the incorporation of the optimal answer at evaluation time for the attention method to see if it would become competitive with the others. Conversely, as a third point, it might be interesting to address only the issue of finding both A and R.

Finally, conjugating both the quality of the approximated rationale and of the generated answer is an interesting challenge leading to a unique and capable model. To do this, reward regularisation, context sampling or compression might be the most straightforward of approaches as has been shown empirically in related works.

6 Acknowledgment

Lize Pirenne gratefully acknowledges the financial support of the Walloon Region for Grant No. 2010235 – ARIAC by DW4AI.

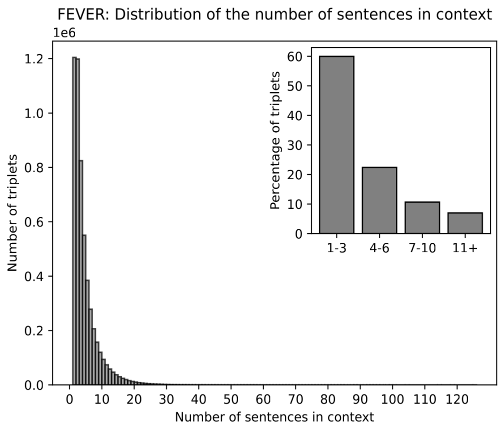

Figure 6: FEVER: Triplet distribution by number of sentences in the context (all wiki pages).

A RDD Dataset

Our enhanced dataset is composed of the textual fields "question", "context", "citation" and "answer" corresponding to (Q_i, C_i, R_i, A_i). These are expanded upon by the fields "sub_question", "sub_citation" and "sub_answer" that list each j triplet (Q_i^j, R_i^j, A_i^j). Each element has its corresponding bounds in "sub_question_index", "sub_citation_index", "sub_answer_index" to avoid searching. The additional fields "id", "num_sub_question" ($\#$("sub_question_index")) and "citation_index" ($\bigcup$"sub_citation_index") are provided for convenience.

If the rationale only contained pronouns, we have chosen to add the closest sentence defining the pronoun. An example of this would be the sentences "Marta is a politician. She is a member of the Green Party." with the question "What party is Marta a member of?" and the answer "Green Party". The rationale would be "Marta is a politician. She is a member of the Green Party.". This was done to avoid ambiguity if only the rationale was presented to a user.

B Dataset comparison

We provide a few figures to illustrate the distribution of the number of sentences in the context for three other datasets: FEVER, MultiRC and Movies. They are respectively shown in Figure 6, Figure 7 and Figure 8.

C IoU, Precision and Recall

We will quickly show that IoU is equivalent to $\frac{1}{\frac{1}{\text{Precision}}+\frac{1}{\text{Recall}}-1}$

The precision is defined as $\frac{TP}{TP+FP}$ and the recall as $\frac{TP}{TP+FN}$ where TP is the number of true positives, FP the number of false positives and FN the number of false negatives. The IoU is defined as $\frac{TP}{TP+FP+FN}$ because predicted $= TP + FP$ and true $= TP + FN$, thus the union of the two is $TP + FP + FN$ and the intersection is TP.

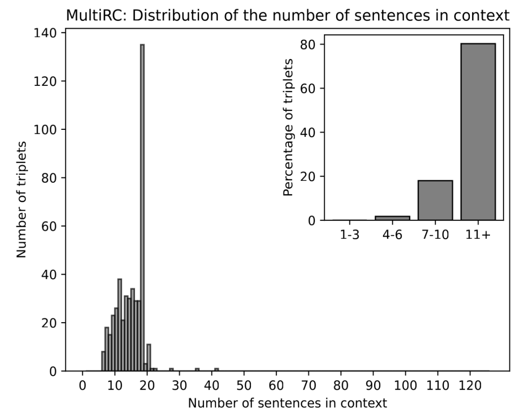

Figure 7: MultiRC: Triplet distribution by number of sentences in the context.

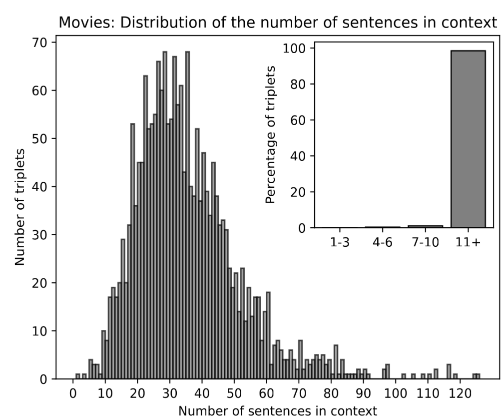

Figure 8: Movies: Triplet distribution by number of sentences in the context.

We can write $\frac{1}{\frac{1}{\text{Precision}}+\frac{1}{\text{Recall}}-1} = \frac{1}{\frac{TP+FP}{TP}+\frac{TP+FN}{TP}-1} = \frac{1}{\frac{TP+FP+TP+FN-TP}{TP}} = \frac{TP}{TP+FP+FN}$.

References

Arous, I.; et al. 2021. MARTA: Leveraging Human Rationales for Explainable Text Classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(7): 5868–5876.

Atanasova, P.; et al. 2022. Diagnostics-Guided Explanation Generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10): 10445–10453.

Banerjee, S.; and Lavie, A. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Goldstein, J.; et al., eds., *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, 65–72. Association for Computational Linguistics.

Bartolo, M.; et al. 2020. Beat the AI: Investigating Adversarial Human Annotation for Reading Comprehension. *Transactions of the Association for Computational Linguistics*, 8: 662–678.

Bastings, J.; and Filippova, K. 2020. The Elephant in the Interpretability Room: Why Use Attention as Explanation When We Have Saliency Methods? *BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*.

Beck, M.; et al. 2024. xLSTM: Extended Long Short-Term Memory. arXiv:2405.04517.

Chae, Y.; and Davidson, T. 2023. Large language models for text classification: From zero-shot learning to fine-tuning. *Open Science Foundation*.

Chalkidis, I.; et al. 2021. Paragraph-level Rationale Extraction through Regularization: A case study on European Court of Human Rights Cases. In Toutanova, K.; et al., eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 226–241. Online: Association for Computational Linguistics.

Chan, A.; et al. 2022. UNIREX: A Unified Learning Framework for Language Model Rationale Extraction. In Chaudhuri, K.; et al., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 2867–2889. PMLR.

Cho, A.; et al. 2024. Transformer Explainer: Interactive Learning of Text-Generative Models. arXiv:2408.04619.

Choi, E.; et al. 2018. QuAC: Question Answering in Context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2174–2184. Brussels, Belgium: Association for Computational Linguistics.

Choromanski, K. M.; et al. 2021. Rethinking Attention with Performers. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Chuang, Y.-S.; et al. 2024. Lookback Lens: Detecting and Mitigating Contextual Hallucinations in Large Language Models Using Only Attention Maps. arXiv:2407.07071.

Clark, J. H.; et al. 2020. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. *Transactions of the Association for Computational Linguistics*, 8: 454–470.

Clark, K.; et al. 2019. What Does BERT Look At? An Analysis of BERT's Attention. arxiv:1906.04341.

Conover, M.; et al. 2023. Free Dolly: Introducing the World's First Truly Open Instruction-Tuned LLM. https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm.

Dasigi, P.; et al. 2019. Quoref: A Reading Comprehension Dataset with Questions Requiring Coreferential Reasoning. arXiv:1908.05803.

De, S.; et al. 2024. Griffin: Mixing Gated Linear Recurrences with Local Attention for Efficient Language Models. arxiv:2402.19427.

Dettmers, T.; et al. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. In Oh, A.; et al., eds., *Advances in Neural Information Processing Systems*, volume 36, 10088–10115. Curran Associates, Inc.

DeYoung, J.; et al. 2020. ERASER: A Benchmark to Evaluate Rationalized NLP Models. arXiv:1911.03429.

Ding, J.; et al. 2023. LongNet: Scaling Transformers to 1, 000, 000, 000 Tokens. arxiv:2307.02486.

Gemma, T.; et al. 2024. Gemma: Open Models Based on Gemini Research and Technology. arxiv:2403.08295.

Glockner, M.; Habernal, I.; and Gurevych, I. 2020. Why do you think that? Exploring Faithful Sentence-Level Rationales Without Supervision. In Cohn, T.; He, Y.; and Liu, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2020*, 1080–1095. Online: Association for Computational Linguistics.

Gu, A.; and Dao, T. 2023. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. arxiv:2312.00752.

Hu, E. J.; et al. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Joshi, B.; et al. 2022. ER-test: Evaluating Explanation Regularization Methods for Language Models. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, 3315–3336. Association for Computational Linguistics.

Khashabi, D.; et al. 2018. Looking Beyond the Surface:A Challenge Set for Reading Comprehension over Multiple Sentences. In *Proceedings of North American Chapter of the Association for Computational Linguistics (NAACL)*.

Kitaev, N.; Kaiser, L.; and Levskaya, A. 2020. Reformer: The Efficient Transformer. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Krishna, S.; et al. 2023. Post Hoc Explanations of Language Models Can Improve Language Models. In Oh, A.; et al., eds., *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Kwiatkowski, T.; et al. 2019. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics*, 7: 453–466.

Lamm, M.; et al. 2021. QED: A Framework and Dataset for Explanations in Question Answering. *Transactions of the Association for Computational Linguistics*, 9: 790–806.

Lewis, P.; et al. 2020a. MLQA: Evaluating Cross-lingual Extractive Question Answering. arXiv:1910.07475.

Lewis, P.; et al. 2020b. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, 9459–9474. Curran Associates, Inc.

Liu, H.; Zaharia, M.; and Abbeel, P. 2023. Ring Attention with Blockwise Transformers for Near-Infinite Context. *CoRR*, abs/2310.01889.

Liu, Y.; et al. 2024. Datasets for Large Language Models: A Comprehensive Survey. arXiv:2402.18041.

Liu, Y.; et al. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692.

Lundberg, S. M.; and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. In Guyon, I.; et al., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ma, X.; et al. 2023. Mega: Moving Average Equipped Gated Attention. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Meng, R.; et al. 2024. SFR-Embedding-Mistral:Enhance Text Retrieval with Transfer Learning. Salesforce AI Research Blog. Accessed: 2024-05-28.

Menick, J.; et al. 2022. Teaching language models to support answers with verified quotes. arXiv:2203.11147.

Moradi, P.; Kambhatla, N.; and Sarkar, A. 2021. Measuring and Improving Faithfulness of Attention in Neural Machine Translation. In Merlo, P.; Tiedemann, J.; and Tsarfaty, R., eds., *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2791–2802. Online: Association for Computational Linguistics.

Muennighoff, N.; et al. 2023. MTEB: Massive Text Embedding Benchmark. arXiv:2210.07316.

Nakayama, H.; et al. 2018. doccano: Text Annotation Tool for Human. https://github.com/doccano/doccano. Accessed: 2024-03-02.

Peng, B.; et al. 2023. RWKV: Reinventing RNNs for the Transformer Era. *Conference on Empirical Methods in Natural Language Processing*.

Rajpurkar, P.; Jia, R.; and Liang, P. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. In Gurevych, I.; and Miyao, Y., eds., *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 784–789. Melbourne, Australia: Association for Computational Linguistics.

Rajpurkar, P.; et al. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Su, J.; Duh, K.; and Carreras, X., eds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392. Austin, Texas: Association for Computational Linguistics.

Reid, M.; et al. 2024. Gemini 1.5: Unlocking Multimodal Understanding across Millions of Tokens of Context. arxiv:2403.05530.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings Using Siamese BERT-networks. *Conference on Empirical Methods in Natural Language Processing*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In Krishnapuram, B.; et al., eds., *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 1135–1144. ACM.

Ross, A. S.; Hughes, M. C.; and Doshi-Velez, F. 2017. Right for the Right Reasons: Training Differentiable Models by Constraining Their Explanations. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 2662–2670. ijcai.org.

Salton, G.; and Buckley, C. 1988. Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing & Management*, 24(5): 513–523.

Sanh, V.; et al. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108.

Shi, F.; et al. 2023. Large Language Models Can Be Easily Distracted by Irrelevant Context. In *Proceedings of the 40th International Conference on Machine Learning*, 31210–31227. PMLR.

Smilkov, D.; et al. 2017. SmoothGrad: removing noise by adding noise. arXiv:1706.03825.

Song, K.; et al. 2023. Zebra: Extending Context Window with Layerwise Grouped Local-Global Attention. arxiv:2312.08618.

Sun, J.; et al. 2022. Investigating the Benefits of Free-Form Rationales. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, 5867–5882. Association for Computational Linguistics.

Sun, X.; et al. 2023. Text Classification via Large Language Models. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, 8990–9005. Association for Computational Linguistics.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic Attribution for Deep Networks. In Precup, D.; and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 3319–3328. PMLR.

Thorne, J.; et al. 2018. FEVER: a Large-scale Dataset for Fact Extraction and VERification. In *NAACL-HLT*.

Wiegreffe, S.; and Marasović, A. 2021. Teach Me to Explain: A Review of Datasets for Explainable Natural Language Processing. arXiv:2102.12060.

Yang, Z.; et al. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yu, M.; et al. 2021. Understanding Interlocking Dynamics of Cooperative Rationalization. In Ranzato, M.; et al., eds., *Advances in Neural Information Processing Systems*, volume 34, 12822–12835. Curran Associates, Inc.

Zaidan, O. F.; Eisner, J.; and Piatko, C. 2008. Machine Learning with Annotator Rationales to Reduce Annotation Cost. In *Proceedings of the NIPS*2008 Workshop on Cost Sensitive Learning*.

Zhai, S.; et al. 2021. An Attention Free Transformer. arxiv:2105.14103.

Zhao, H.; et al. 2024. Explainability for Large Language Models: A Survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2).