# Enhancing LLMs via Lightweight Question-Attended Span Extraction

**Anonymous ACL submission**


## Abstract

To address the hallucination challenge in zero-shot LLMs without extensive task-specific prompt engineering, we introduce a lightweight **Q**uestion-**A**ttended **S**pan **E**xtraction (*QASE*) module during the fine-tuning of LLMs. Our experiments demonstrate that *QASE* empowers smaller models to outperform SOTA LLMs on reading comprehension tasks, notably achieving up to a 32.6% improvement over GPT-4's F1 score on SQuAD, all without increasing computational costs.[1]


## 1 Introduction

The rapid progress of large language models (LLMs) like GPT-4 (OpenAI, 2023), Llama 2 (Touvron et al., 2023), and PaLM 2 (Anil et al., 2023) has garnered much attention. Yet these powerful models face the challenge of hallucination (Ji et al., 2023; Bang et al., 2023), where incorrect or fabricated information is generated. Techniques like prompt engineering can mitigate this to some extent, but more work is needed for broader applicability (White et al., 2023). Fine-tuning these models for downstream tasks is costly due to their size, although efforts like Alpaca-LoRA (Hu et al., 2021) attempt to reduce computational costs.

In this paper, we address hallucination in pre-trained LLMs (PLMs) using a lightweight **Q**uestion-**A**ttended **S**pan **E**xtraction (*QASE*) module. We conduct experiments on reading comprehension datasets to evaluate its effectiveness in enhancing LLMs to generate context-grounded answers. Our contributions include:

1. Developing *QASE*, a lightweight module, enabling smaller models to outperform SOTA LLMs on MRC tasks, notably surpassing GPT-4 on SQuAD by up to 32.6% on F1 score.
2. *QASE* boosts performance without increasing computational costs, aiding researchers with limited resources.

[1] Our code is available at this anonymous repo link.

## 2 Related Work

**Machine Reading Comprehension** (MRC) is a notable challenge in NLP. In recent years, many MRC benchmark datasets have been created, including typical question/answer corpora like SQuAD (Rajpurkar et al., 2016), and more complex question/multi-span answer corpora such as Quoref (Dasigi et al., 2019) and MultiSpanQA (Li et al., 2022).

Most current studies approach the MRC task by predicting the start and end positions of the answer spans from a given context (Ohsugi et al., 2019; Lan et al., 2019; Bachina et al., 2021). To handle the multi-span setting, some studies frame the problem as a sequence tagging task (Hu et al., 2019; Segal et al., 2020), while others explore ways to combine models with different tasks (Lee et al., 2023; Zhang et al., 2023).

Work most similar to ours focuses on using the power of generative-based language models (Yang et al., 2020; Li et al., 2021; Su et al., 2022). However, there is little research on using the emerging abilities of LLMs for MRC tasks.

## 3 Method

### 3.1 Question-Attended Span Extraction

To address the hallucination problem in generative models, we incorporate our question-attended span extraction mechanism, *QASE*, during the fine-tuning of the models. This mechanism ensures that generated answers are grounded in the original provided context. We cast span extraction as a sequence tagging problem and employ the Inside-Outside (IO) tagging schema, where each token in the sequence is tagged as 'inside' (***I***) if it is part of a relevant span, or 'outside' (***O***) if it is not. This schema generalizes well to both single- and multi-span extraction settings, achieving comparable or even better performance than the well-known BIO tagging format (Huang et al., 2015), as shown by Segal et al. (2020).

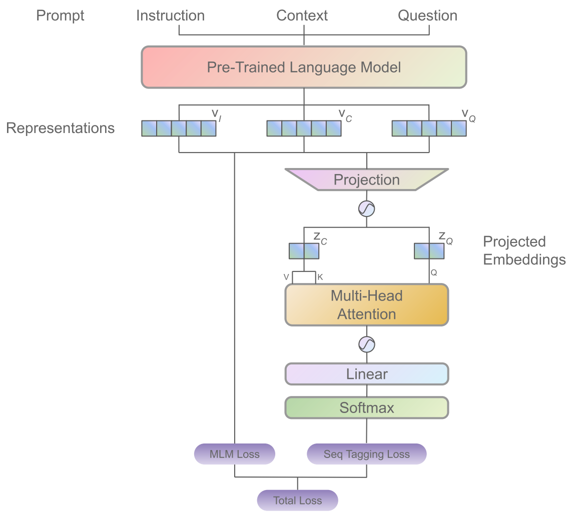


Figure 1: Overview of Our Model

The overall architecture of our proposed model is shown in Figure 1. Given input context $C$ and question $Q$, we first concatenate these together with a separator for delineation and then feed them into the generative language model. The hidden states from the language model are then passed through projection layers to produce embeddings $z_i = ReLU(W_{proj} v_i + b_{proj})$, where $h_i \in R^d$ is the output hidden state of the language model for the $i^{th}$ token.

To learn representations of each context token based on specific questions, we employ a **multi-head attention** mechanism ($MHA$). Each head in $MHA$ attends to different aspects of the context in relation to the question, utilizing question embeddings as the query and context embeddings as the key and the value. This mechanism enhances the model's understanding and response generation by grounding the context token representations in the specifics of the queried question. The projected embeddings $z_i$ of the $i^{th}$ are passed through the multi-head attention component, and subsequently channeled through a linear layer and a softmax layer to compute the probability:

$$p_i = softmax(W_{lin} \cdot MHA(z_i) + b_{lin}) \quad (1)$$

which denotes the probability of the $i^{th}$ token being inside the answer spans. We then compute the sequence tagging loss using the cross entropy loss:

$$L_{QASE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=0}^{1} y_{ij} log(p_{ij}) \quad (2)$$

where $j \in 0, 1$ corresponds to class $\boldsymbol{O}$ and class $\boldsymbol{I}$, and $y_{ij}$ is a binary value indicating whether the $i^{th}$ token belongs to class $j$.

### 3.2 Joint MLM Fine-Tuning

We fine-tune PLMs using multi-task learning, training concurrently on both the masked language modeling loss and sequence tagging loss: $L = L_{MLM} + \beta L_{QASE}$, where $\beta$ is a hyper-parameter that controls the weight of the span extraction task. This approach refines the PLMs to become adept at generating answers that are grounded in the original context, effectively targeting the hallucination issue which has been common in recent LLMs.

## 4 Experiments

### 4.1 Datasets and Metrics

Given our objective of generating context-grounded answers, we utilize the following three datasets.

**MultiSpanQA** (Li et al., 2022): This reading comprehension dataset consists of over 6.5k question-answer pairs. Unlike most existing single-span answer MRC datasets, MultiSpanQA focuses on multi-span answers.

**SQuAD** (Rajpurkar et al., 2016): A benchmark reading comprehension dataset consisting of 100K+ questions with single-span answers. We use SQuAD v1.1. Since the official evaluation on v1.1 has long been terminated, we report our results on the official v1.1 development set.

**Quoref** (Dasigi et al., 2019): A benchmark reading comprehension dataset containing more than 24K questions, with the majority of answers being single-span, and approximately 10% being multi-span.

For MultiSpanQA, we employ the exact match (EM) and partial match (Overlap) F1 scores as metrics, following the conventions of its official leaderboard. For SQuAD and Quoref, we use the exact match percentage and macro-averaged F1 score as metrics.

### 4.2 Experimental Setup

To evaluate the effectiveness of our *QASE* component independent of any specific language model, we experiment with multiple open-source LLMs. These include both decoder-only LLMs, such as **Llama 2** (Touvron et al., 2023) and **Alpaca** (Taori et al., 2023), and an encoder-decoder model, **Flan-T5** (Chung et al., 2022).

For Llama 2, we fine-tune the pre-trained 7B version using LoRA (Hu et al., 2021) and instruction-tuning. During fine-tuning, both with and without *QASE*, we incorporate instructions into the prompt to explicitly instruct the model to generate answers "with exact phrases from the context and avoid explanations." The same setup is used for fine-tuning the pre-trained Alpaca model. For the family of Flan-T5 models, we fine-tune the small, the base, and the large versions. The number of trainable parameters for each model is provided in Table 1.

| | **Trainable Params** |
|---|---|
| Llama 2/Alpaca LoRA | 4,194,304 |
| Flan-T5-Small | 76,961,152 |
| Flan-T5-Base | 247,577,856 |
| Flan-T5-Large | 783,094,784 |
| *QASE* | 1,314,306 ∼ 3,149,314 |

Table 1: Trainable parameters of experimented models.

We train all our models on single GPUs, using a batch size of 2-4 depending on the VRAM of the respective GPUs. We use four types of GPUs: A40, A10, A5500, and A100. Notably, the Flan-T5-Large model can only be accommodated on the A100 GPU due to its demanding resources. Models are trained for 3 epochs or until convergence.

### 4.3 Model Comparisons

We compare the zero-shot performance of various PLMs to that of their corresponding versions fine-tuned with our proposed *QASE* component. The results, presented in Table 6 in Appendix A.2, show that fine-tuning with *QASE* improves performance across all datasets. Specifically, on the MultiSpanQA dataset, models using *QASE* perform up to 124.4 times better in exact match and 3.4 times better in F1 score compared to the original models. On the SQuAD dataset, the exact match improves by up to 5.6 times, and F1 score by up to 3.0 times. Similarly, on the Quoref dataset, the exact match improves by up to 38.4 times, and F1 score by up to 11.2 times with *QASE*.

We further compare our best performing model, Flan-T5-Large$_{QASE}$, with SOTA models, alongside zero-shot GPT-3.5 and GPT-4. GPT-3.5 is the most capable and cost-effective model in the OpenAI GPT family, and GPT-4 exhibits even more advanced reasoning capabilities (Liu et al., 2023b). Research suggests that both outperform the traditional fine-tuning method on most logical reasoning benchmarks (Liu et al., 2023a).

On MultiSpanQA, we compare Flan-T5-Large$_{QASE}$ with GPT variants and models on the official MultiSpanQA leaderboard, as referred to in Appendix A.1. Figure 2 and Table 2 show that Flan-T5-Large$_{QASE}$ outperforms LIQUID (Lee et al., 2023), which currently ranks #1 on the leaderboard, with respect to the overlap F1 score. Moreover, it surpasses GPT-4 by 4.5% on the exact match F1 and 1.5% on the overlap F1.

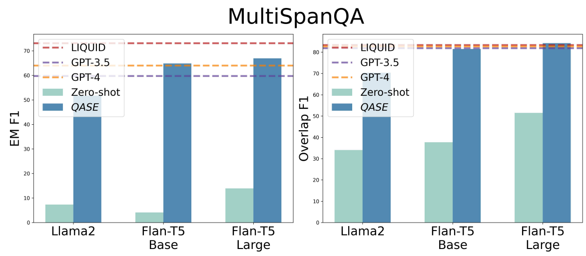


Figure 2: Performance of zero-shot PLMs, GPTs, SOTA, and $QASE$ fine-tuned PLMs on **MultiSpanQA** test set.

| | **EM F1** | **Overlap F1** ↑ |
|---|---|---|
| Flan-T5-Large | 13.907 | 51.501 |
| Flan-T5-Base$_{QASE}$ | 64.874 | 81.498 |
| GPT-3.5 | 59.766 | 81.866 |
| GPT-4 | 64.027 | 82.731 |
| LIQUID (Lee et al., 2023) | **73.130** | 83.360 |
| Flan-T5-Large$_{QASE}$ | 66.918 | **84.221** |

Table 2: Performance comparison between baselines and Flan-T5-Large$_{QASE}$ on the **MultiSpanQA** test set.

For SQuAD, we compare Flan-T5-Large$_{QASE}$ with GPT variants and models on the official SQuAD v1.1 leaderboard, as referred to in Appendix A.1. Figure 3 and Table 3 show that Flan-T5-Large$_{QASE}$ surpasses human performance, equaling the performance of the NLNet model from Microsoft Research Asia and the original pre-trained BERT-Large from Google (Devlin et al., 2019), which are ranked #11 and #13 on the v1.1 leaderboard respectively. Additionally, it surpasses GPT-4 by 113.8% on the exact match score and 32.6% on F1.

For Quoref, we compare Flan-T5-Large$_{QASE}$ with GPT variants and models on the official Quoref leaderboard, as referred to in Appendix A.1. As shown in Figure 4 and Table 4, Flan-T5-Large$_{QASE}$ is comparable to CorefRoberta-Large (Ye et al., 2020), which ranks #9 on the leaderboard,

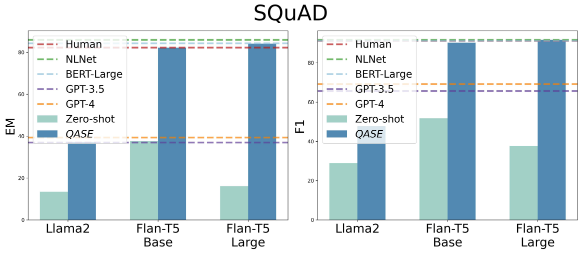


Figure 3: Performance of zero-shot PLMs, GPTs, SOTA, and $QASE$ fine-tuned PLMs on **SQuAD** test set.

| | **EM** | **F1** ↑ |
|---|---|---|
| Flan-T5-Large | 16.149 | 37.691 |
| GPT-3.5 | 36.944 | 65.637 |
| GPT-4 | 39.347 | 69.158 |
| Flan-T5-Base$_{QASE}$ | 82.204 | 90.240 |
| Human Performance | 82.304 | 91.221 |
| BERT-Large (Devlin et al., 2019) | 84.328 | 91.281 |
| MSRA NLNet (ensemble) | **85.954** | 91.677 |
| Flan-T5-Large$_{QASE}$ | 84.125 | **91.701** |

Table 3: Performance comparison between baselines and Flan-T5-Large$_{QASE}$ on the **SQuAD** dev set.

with a 0.5% higher exact match. Furthermore, it outperforms GPT-4 by 11.9% on the exact match and 4.8% on F1.

These results show that, by employing *QASE*, generative-based PLMs can be fine-tuned to produce high-quality, context-grounded answers in reading comprehension tasks. This fine-tuning mechanism enables them to match the performance of SOTA models, which are typically optimized for span boundary detection or sequence tagging objectives, especially in the multi-span setting.

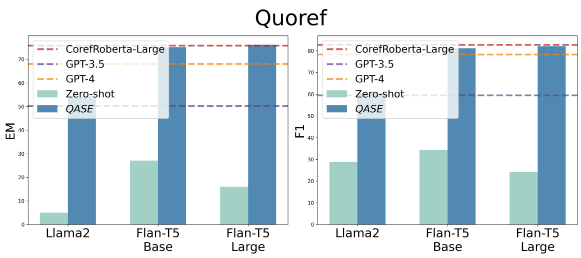


Figure 4: Performance of zero-shot PLMs, GPTs, SOTA, and $QASE$ fine-tuned PLMs on **Quoref** test set.

### 4.4 Ablation Studies

We conducted ablation studies to assess the contribution of the *QASE* component in fine-tuning the PLMs. Table 5 reports the F1 scores of models fine-tuned with and without *QASE*. Model variants derived from the same base PLM, fine-tuned both with and without *QASE*, share identical configurations including learning rate, weight decay, batch size, epoch number, and GPU type. Overall, models fine-tuned with *QASE* consistently outperform their counterparts fine-tuned without *QASE*. Specifically, on MultiSpanQA, models with *QASE* exhibit a performance improvement of up to 3.3% compared to vanilla fine-tuned models. On SQuAD, the F1 score improves by up to 8.4%. Similarly, on Quoref, the F1 score is enhanced by up to 16.0%.

The results of the ablation studies demonstrate that our proposed *QASE* component is effective in enhancing the performance of fine-tuned generative-based PLMs, enabling them to produce high-quality context-grounded answers.

| | **EM** | **F1** ↑ |
|---|---|---|
| Flan-T5-Large | 15.96 | 24.10 |
| GPT-3.5 | 50.22 | 59.51 |
| GPT-4 | 68.07 | 78.34 |
| Flan-T5-Base$_{QASE}$ | 75.17 | 81.18 |
| CorefRoberta-Large (Ye et al., 2020) | 75.80 | **82.81** |
| Flan-T5-Large$_{QASE}$ | **76.19** | 82.13 |

Table 4: Performance comparison between baselines and Flan-T5-Large$_{QASE}$ on the **Quoref** test set.

| | **MultiSpanQA** | **SQuAD** | **Quoref** |
|---|---|---|---|
| Llama2$_{FT}$ | 68.140 | 47.055 | 52.09 |
| Llama2$_{QASE}$ | **70.389** | **47.686** | **60.44** |
| Alpaca$_{FT}$ | 69.099 | 43.950 | - |
| Alpaca$_{QASE}$ | **70.008** | **47.622** | - |
| Flan-T5-Small$_{FT}$ | 76.494 | 85.513 | 63.30 |
| Flan-T5-Small$_{QASE}$ | **77.103** | **85.901** | **66.88** |
| Flan-T5-Base$_{FT}$ | 81.408 | 89.558 | 80.90 |
| Flan-T5-Base$_{QASE}$ | **81.498** | **90.240** | **81.18** |
| Flan-T5-Large$_{FT}$ | 83.094 | 90.712 | 80.49 |
| Flan-T5-Large$_{QASE}$ | **84.221** | **91.701** | **82.13** |

Table 5: Performance (F1) of fine-tuned (*FT*) PLMs without and with *QASE*.

## 5 Conclusion

In this study, we address hallucinated text generation in pre-trained LLMs using *QASE*, a lightweight question-attended span extraction module, during fine-tuning. *QASE* enhances smaller models to outperform GPT-4 on all three MRC datasets by significant margins in exact match and F1 scores. Utilizing *QASE*, Flan-T5-Large models match the performance of leading non-generative MRC models optimized for span detection or tagging, even surpassing the top-ranked SOTA model on the MultiSpanQA leaderboard. Importantly, *QASE* improves performance without additional computational costs, providing an economic solution for researchers with more limited resources.

## Limitations

Due to our limited computational resources, we have been able to perform our experiments on models no larger than Flan-T5-Large. This same constraint led us to only fine-tuning of Llama 2 and Alpaca with LoRA. We note that models based on Llama 2 and Alpaca generally underperform those based on Flan-T5. Apart from the inherent distinctions between decoder-only and encoder-decoder models, and their suitability for different tasks (as seen from the models' zero-shot performance), a possible factor could be the number of trainable parameters during fine-tuning. Specifically, fine-tuning Llama 2 and Alpaca with LoRA results in only 4M trainable parameters, while even the smallest Flan-T5 model provides 76M trainable parameters. We acknowledge that many researchers face similar computational resource limitations. Therefore, our research should be very useful, proposing this lightweight module capable of enhancing smaller models to outperform SOTA LLMs on MRC tasks like these, achieving a balance of effectiveness and affordability.

## References


Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Sony Bachina, Spandana Balumuri, and Sowmya Kamath S. 2021. Ensemble ALBERT and RoBERTa for span prediction in question answering. In *Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering (DialDoc 2021)*, pages 63–68, Online. Association for Computational Linguistics.

Y Bang, S Cahyawijaya, N Lee, W Dai, D Su, B Wilie, H Lovenia, Z Ji, T Yu, W Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. arxiv.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Pradeep Dasigi, Nelson F. Liu, Ana Marasović, Noah A. Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5925–5932, Hong Kong, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Minghao Hu, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019. A multi-type multi-span network for reading comprehension that requires discrete reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1596–1606, Hong Kong, China. Association for Computational Linguistics.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. arxiv 2015. *arXiv preprint arXiv:1508.01991*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Seongyun Lee, Hyunjae Kim, and Jaewoo Kang. 2023. Liquid: A framework for list question answering dataset generation. *arXiv preprint arXiv:2302.01691*.

Chenliang Li, Bin Bi, Ming Yan, Wei Wang, and Songfang Huang. 2021. Addressing semantic drift in generative question answering with auxiliary extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 942–947, Online. Association for Computational Linguistics.

Haonan Li, Martin Tomko, Maria Vasardani, and Timothy Baldwin. 2022. MultiSpanQA: A dataset for multi-span question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1250–1260, Seattle, United States. Association for Computational Linguistics.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023a. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*.

Xiao Liu, Junfeng Yu, Yibo He, Lujun Zhang, Kaiyichen Wei, Hongbo Sun, and Gang Tu. 2023b. System report for CCL23-eval task 9: HUST1037 explore proper prompt strategy for LLM in MRC task. In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 3: Evaluations)*, pages 310–319, Harbin, China. Chinese Information Processing Society of China.

Yasuhito Ohsugi, Itsumi Saito, Kyosuke Nishida, Hisako Asano, and Junji Tomita. 2019. A simple but effective method to incorporate multi-turn context with BERT for conversational machine comprehension. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 11–17, Florence, Italy. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Elad Segal, Avia Efrat, Mor Shoham, Amir Globerson, and Jonathan Berant. 2020. A simple and effective model for answering multi-span questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3074–3080, Online. Association for Computational Linguistics.

Dan Su, Xiaoguang Li, Jindi Zhang, Lifeng Shang, Xin Jiang, Qun Liu, and Pascale Fung. 2022. Read before generate! faithful long form question answering with machine reading. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 744–756, Dublin, Ireland. Association for Computational Linguistics.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

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*.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*.

Junjie Yang, Zhuosheng Zhang, and Hai Zhao. 2020. Multi-span style extraction for generative reading comprehension. *arXiv preprint arXiv:2009.07382*.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential Reasoning Learning for Language Representation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7170–7186, Online. Association for Computational Linguistics.

Chen Zhang, Jiuheng Lin, Xiao Liu, Yuxuan Lai, Yansong Feng, and Dongyan Zhao. 2023. How many answers should i give? an empirical study of multi-answer reading comprehension. *arXiv preprint arXiv:2306.00435*.

# A Appendix

## A.1 Dataset Leaderboard

Below are the official dataset leaderboards we refer to:

- **MultiSpanQA**: https://leaderboard.allenai.org/quoref/submissions/public
- **SQuAD**: https://rajpurkar.github.io/SQuAD-explorer/
- **Quoref**: https://leaderboard.allenai.org/quoref/submissions/public

## A.2 Experiment Results

Below are the complete results from all our experiments:

| | MultiSpanQA | | SQuAD | | Quoref | |
|---|---|---|---|---|---|---|
| | **EM F1** | **Overlap F1** | **EM** | **F1** | **EM** | **F1** |
| Llama2 | 7.354 | 34.031 | 13.443 | 28.931 | 5.02 | 28.91 |
| Llama2$_{FT}$ | 50.934 | 68.140 | 36.679 | 47.055 | 45.52 | 52.09 |
| Llama2$_{QASE}$ | **51.748** | **70.389** | **37.219** | **47.686** | **54.28** | **60.44** |
| Alpaca | 15.201 | 42.759 | 18.259 | 33.871 | - | - |
| Alpaca$_{FT}$ | **52.730** | 69.099 | 27.881 | 43.950 | - | - |
| Alpaca$_{QASE}$ | 52.196 | **70.008** | **37.313** | **47.622** | - | - |
| Flan-T5-Small | 0.475 | 22.539 | 13.878 | 28.710 | 1.58 | 5.96 |
| Flan-T5-Small$_{FT}$ | **59.128** | 76.494 | 77.332 | 85.513 | 58.21 | 63.30 |
| Flan-T5-Small$_{QASE}$ | 59.080 | **77.103** | **77.663** | **85.901** | **60.70** | **66.88** |
| Flan-T5-Base | 4.113 | 37.694 | 37.596 | 51.747 | 27.08 | 34.38 |
| Flan-T5-Base$_{FT}$ | 64.659 | 81.408 | 82.090 | 89.558 | 72.77 | 80.90 |
| Flan-T5-Base$_{QASE}$ | **64.874** | **81.498** | **82.204** | **90.240** | **75.17** | **81.18** |
| Flan-T5-Large | 13.907 | 51.501 | 16.149 | 37.691 | 15.96 | 24.10 |
| Flan-T5-Large$_{FT}$ | **67.408** | 83.094 | 83.159 | 90.712 | 75.17 | 80.49 |
| Flan-T5-Large$_{QASE}$ | 66.918 | **84.221** | **84.125** | **91.701** | **76.19** | **82.13** |

Table 6: Performance of zero-shot PLMs and fined-tuned PLMs with and without *QASE*.