Empowering Multi-Span Question Answering with Expansive Information Injection using Large Language Models

Zhiyi Luo, Yingying Zhang, Ying Zhao, Shuyun Luo* and Wentao Lv

[a] *School of Computer Science and Technology and the Key Laboratory of Intelligent Textile and Flexible Interconnection of Zhejiang Province, Zhejiang Sci-Tech University*

[b] *No. 928, No. 2 street, Baiyang street, Qiantang New District, Hangzhou 310018, China*

ABSTRACT

Multi-span question answering has gained prominence as it aligns more closely with real-world user requirements compared to single-span question answering. The utilization of pretrained language models has shown promise in improving multi-span question answering, particularly for factoid questions that necessitate entity-based answers. However, existing methods tend to overlook critical information regarding answer span boundaries, resulting in limited accuracy when generating descriptive answers. To address this limitation, we propose TOAST, a novel joint learning framework specialized in token-based neighboring transitions that capture answer span boundaries through adjacent word relations. Our approach extracts high-quality multi-span answers and is general-purpose, applicable to both alphabet languages like English and logographic languages like Chinese. Furthermore, we introduce CLEAN, a comprehensive open-domain Chinese multi-span question answering dataset, which includes a substantial number of descriptive questions. Extensive experiments demonstrate the superior performance of TOAST over previous top-performing QA models in terms of both EM F1 and overlapped F1 scores. Specifically, the TOAST models, leveraging BERT_{base} and RoBERTa_{base}, achieve substantial improvements in EM F1 scores, with increments of 3.03/2.13, 4.82/3.73, and 16.26/11.53, across three publicly available datasets, respectively.

1. Introduction

Let's cite a paper **?**.

In summary, the main contributions in this paper are as follows:

- We employs a automatic data augmentation framework using Large Language Model (***LLM***) as a knowledge source and a extra content supplement to linearize relevant information and possible continuation from LLM as texts, then inject them into original contexts.

- We develop a series of prompt templates designed for interacting with ChatGPT to acquire comprehensive explanations of numerous entities. These templates ensure that the formats of the responses provided by ChatGPT are highly parseable and well-structured.

2. relation

3. Our Approach

Existing extractive multi-span question answering models exhibit subpar performance in handling ambiguous words, complex proper nouns (such as film and song titles), numerical values, and long descriptive answers. The main reasons for these shortcomings can be categorized into three aspects:

Ambiguous Words: The real world is replete with words that have a single form but multiple meanings, which heavily depend on the context. For instance, the word "Cameron" could refer to a famous director or a former British Prime Minister, depending on the context. The specific meanings of such words often appear infrequently in training corpora, making them difficult to learn.

Numerical Values: In contrast to ambiguous words, numerical values have a single meaning but can be represented in various forms. For example, "22.5 billion years" can also be expressed as "cosmic years".

Multi-span Answers: In extractive multi-span question answering tasks, the model needs to grasp the overall relationship of multi-span answers in the context, such as parallelism and progression.These challenges require the model to possess a high level of comprehensive understanding ability and knowledge about language and the world.

Leveraging large language models to parse question-answering data and inject auxiliary knowledge into the model can promote the integration of the model's latent world knowledge and specific domain knowledge in the paragraph, thereby enhancing the model's performance. Thus, we use GPT-3.5-turbo to generate entity annotations, entity association analysis, and content continuation for each question-answering paragraph as auxiliary information for the model:

Context Supplementary: By inserting explanations of entities in the form of annotations after the

*Corresponding author

luozhiyi@zstu.edu.cn (Z. Luo); 272831920@qq.com (Y. Zhang); 308956149@qq.com (Y. Zhao); shuyunluo@zstu.edu.cn (S. Luo); alvinlwt@zstu.edu.cn (W. Lv)

http://zhiyiluo.site (Z. Luo)

ORCID(s): 0000-0002-2206-1926 (Z. Luo)

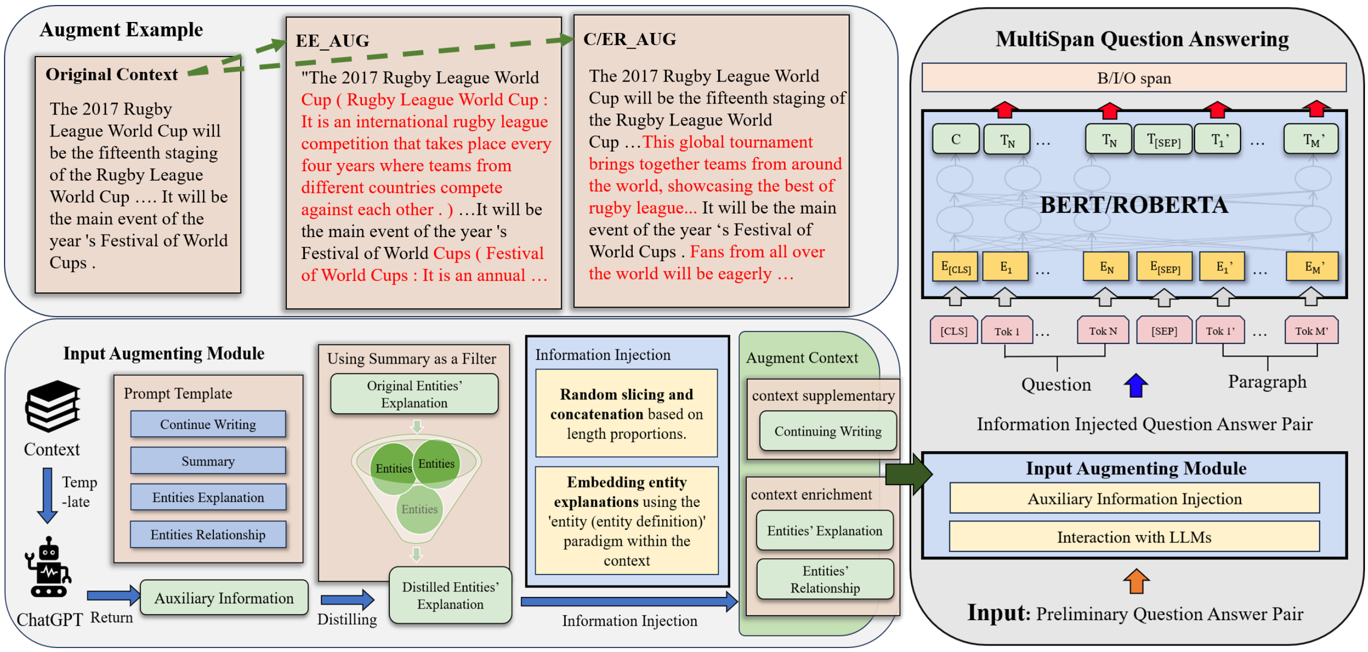

Figure 1: An overview of our automatic information augmentation framework. **(a) Step 1**: Interact with ChatGPT to Get Auxiliary Information. **(b) Step 2**: Distilling the Information and Injecting Them into Context **(c) Step 3**: Input the Augmented Context into Tagging Model

main entities or concept words in the question and paragraph, this method helps the model capture the actual meaning of ambiguous words in specific contexts. It provides direct information prompts for low-frequency meanings or low-frequency words, achieving entity or concept alignment in the question-answering system.

Context Enrichment: By integrating entity association analysis and content continuation with the original question-answering text to form new paragraphs based on knowledge enhancement, this method captures the inherent associations between words or entity concepts with the same meaning but different forms. On one hand, it parses the logical relationships between other entities using entity relationship analysis. On the other hand, it extends the original paragraph information using content continuation, introducing more external knowledge while helping the model understand the overall content direction of the paragraph, thereby enhancing the model's comprehension ability.

Overall,we design an automated knowledge enhancement method for multi-span question answering tasks, which interacts with large language models based on templates. This method is applicable to all question answering models or frameworks. It is universally applicable to any other downstream tasks that contain paragraph data, without the need for any complex calculations. In the following sections, we will provide a detailed introduction to each specific module of this process.

3.1. Construction of Prompt Template

The quality and format of the content produced by LLMs depends greatly on the prompts. Hence, it is necessary to clarify the format and content requirements of the outputs in advance, and to refine the templates that meet the expectations through extensive evaluation. For entity relationship parsing and textual continuation, it would be better to ensured that the the output is a natural paragraph, to facilitate the automatic splicing of the enriched knowledge with the initial paragraphs,which is consistent with encoder's natural input format. Regarding knowledge of entity explanation, the inserted knowledge fusion method requires a parsable and structured outputs for automatic recognition and segmentation of entities and their explanations in post-processing, as well as inserting explanation behind the description of matched entity in initial context.

Concretely, by clearly specifying the form and requirements in the prompts, especially for entity relationship parsing and contextual continuation, which need to be formulated in the form of natural statements, the targeted enhanced knowledge can be obtained more easily. Furthermore, to obtain well-structured results for complex model interactions, like entity explanations, we utilize a set of corresponding hint templates to construct the process, which has proven to be useful for interactions that require semi-structured answers through verification.

Eventually, the template is piloted to clarify any ambiguous requirements in the template to confirm an accurate understanding of the LLMs (e.g., adding at

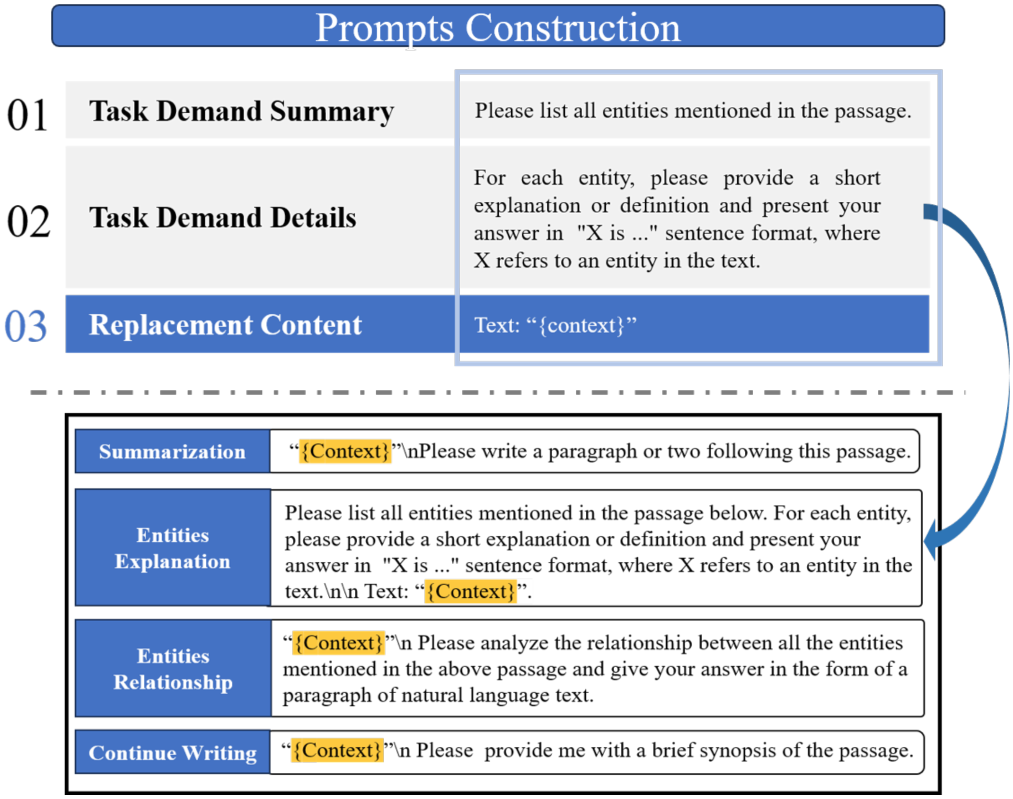

Figure 2: Prompt Templates and it's Construction

the end of the prompt, "Please return the answer in the following form, being careful not to repeat it: \n\nA is .....") In addition, to avoid omission or confusion of requirements due to the length of the prompt text, specific principles should be reiterate in a separate paragraph.

3.2. Information Injection

To incorporate enhanced knowledge into the original data (comprising questions and paragraphs) pertaining to entity interpretation, a methodology involving the insertion of explanations is employed. Specifically, the process begins with the automated script parsing of entity interpretation data. This script, on one hand, utilizes regular expressions to analyze the structure of the majority of sentences, while on the other hand, for a limited subset of sentences that deviate from the template specifications, employs Spacy's Semantic Dependency Analysis model to identify entities and their corresponding interpretations.

Upon obtaining the "Entity-Entity Interpretation" knowledge base for each question-and-answer data, a subsequent scan of the original context is conducted. For each entity that exists in the dataset's library of entities and their explanations, the corresponding explanation is inserted immediately following the entity, enclosed within parentheses. This meticulous approach ensures that the augmented text, enriched with knowledge, retains its natural sentence or paragraph flow. For instance, a resulting sentence might appear as follows: "How long does it take for the Milky Way (Milky Way: The galaxy that contains our Solar System and is home to billions of stars.) to rotate?"

For the incorporation of enhanced knowledge related to entity relationship analysis and content continuation, a method of random slicing and concatenation based on length ratios is employed to infuse the augmented knowledge into the original paragraphs. Initially, the ratio of the average length of the augmented knowledge text to that of the original paragraph is calculated. Subsequently, at the model's input layer, the original input data is segmented into text fragments of length of *512 * average original length / (average original length + average augmented text length)*, and the augmented text is also segmented into text fragments of length of *512 * average augmented text length / (average original length + average augmented text length)*. Then, random selections of augmented text fragments are made and concatenated into the middle of each original text fragment. Finally, this concatenated longer text is fed into the model in a sliding manner to ensure comprehensive interaction between the original data and the augmented data at lower layers of the model, as illustrated in Figure 4.

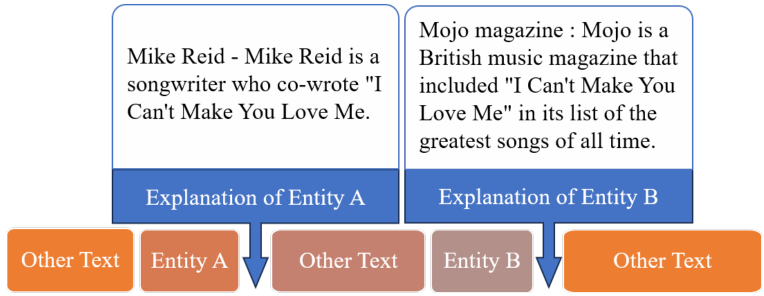

Figure 3: The Process of Inserting Entity Explanation into Context

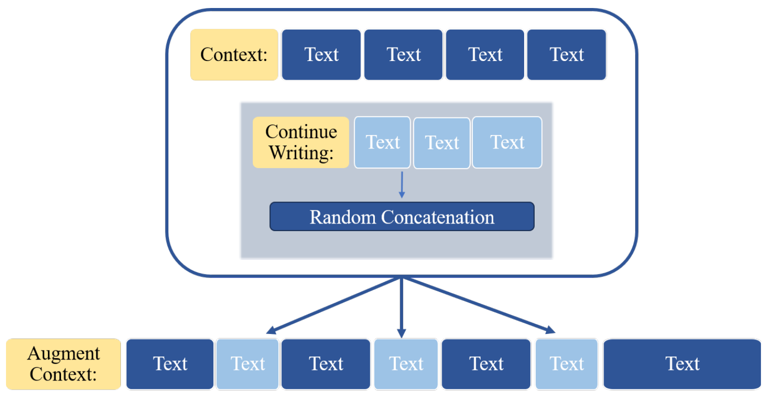

Figure 4: The Process of Concatenation of Original Context and Auxiliary Information

3.3. Information Integration

In addition to the aforementioned three types of information, we seek more valuable enhanced information by requesting summaries for each question-context pair from Large Language Models (LLMs). We then extract entities mentioned in the summaries to capture those highly relevant to the core context. Moreover, recognizing the intrinsic connection between entity explanation and entity relationship parsing, we categorize them as complementary information for context enrichment, with context continuation serving as supplementary information. This approach aims to augment the model's internal text-parsing knowledge with external world knowledge through enhanced information.

To further investigate the efficacy of multi-information integration, leveraging enhanced information derived from Large Language Models (LLMs), we employ two distinct approaches to combine entity knowledge and context continuity. The first approach involves the incorporation of all information within the original context, with each type of information undergoing individualized knowledge injection. The second approach, referred to as "bagging," treats models trained on diverse information sources as voters. For each query, we employ all model predictions for token-level BIO labeling within the context, determining the final prediction through a majority vote. In cases of tie-breaking, we prioritize the logits' magnitude to ascertain the ultimate prediction.

4. Experiments

In this section, we compare our information augmentation approach with multiple strong baseline on multi-span question answering. We first introduce the datasets and experiment setup, then show the experimental results and analysis for different model.

4.1. Evaluation Dataset

We conducted experiments on MultiSpanQA(Li et al., 2022), a recently introduced Reading Comprehension dataset designed for multi-span question answering. This dataset comprises 6.5K multi-span examples in which the questions represent user queries issued to the Google search engine, and the contexts are extracted from the English Wikipedia. It's worth noting that there is a expand variant of MultiSpanQA known as MultispanQA(expand), which intakes single-span and answerable questions. However, we did not perform a comparison with the expanded dataset due to its relatively lower proportion of multi-span QA pairs.

4.2. Experimental Setup

For all competing models and our model, we use the HuggingFace implementation of BERT$_{base}$ or RoBERTa$_{base}$ as the *encoder* with $max\_len = 512$. We set the initial learning rate as 3×10^{-5} and $batch\_size = 4$, and use the BERTAdam optimizer with a weight decay of 0.01. Our approach does not involve tuning the parameters on the validation set. Instead, we rely on the model checkpoints obtained after 5 epochs. Next, we introduce the comparison model and evaluation metrics in our experiments.

4.2.1. Model Under Comparison

We introduce two comstracting models approaches to multi-span answer extraction : **TASE** (Segal et al., 2020) and **LIQUID**(Lee et al., 2023). TASE utilizes a tag-based span extraction model which identifies multi-span answers though the assigning a tag to every input token with BIO tagging scheme. On the other hand, LIQUID serves as a framework for generating multi-span QA datasets to improve model performance.

To enhance the context with auxiliary information, we employ two distinct approaches: $\mathbf{AUG}_c$ and $\mathbf{AUG}_{eree}$, where **AUG** is our automatic data augmentation framework,and the suffix indicates which kind of information is injected into the context. $\mathbf{AUG}_C$ enriches the context with continue writing, while $\mathbf{AUG}_{EREE}$ supplements context with entities information including explanation and relationship analysis. Specifically, we leverage ChatGPT as a knowledge source to linearize the relevant information from large language models. in texts format and seamlessly integrate into the original contexts, thus reinforces the information of model inputs.

4.2.2. Evaluation Metrics

We use two automatic metrics for evaluation: Exact Match and Overlap F1 score.

- **Exact Match**. An exact match occurs when a predicted span fully matches one of the ground-truth answer spans. We calculate the micro-average precision, recall and f1 score for the extract match metric.
- **Overlap F1 score**. Overlap F1 score is the macro-average f1 score, where the f1 score for each example is computed by treating the prediction and gold as a bag of tokens.

4.3. Experimental Results and Analysis

In this section, we compare AUG with all competing models described above quantitatively.

4.3.1. Comparison Results

We evaluate our model as well as baselines (*Section* 4.2.1) on the development splits of multi-span datasets (*Section* 4.1) using automatic metrics (*Section* 4.1). The comparison results are shown in Table 1, Table 2.

Table 1 and Table 2 illustrate the performance comparison between the proposed approaches, AUG_c and AUG_{eree}, and several strong baselines, including the previous state-of-the-art model LIQUID. These comparisons are conducted using both the BERT_{base} and RoBERTa_{base} encoders, and regard multi-span question answering as a *BIO* sequence tagging task to predict each token whether it is a part or begin of an answer. Notably,AUG_c exhibit superior performance across the evaluate dataset on all metrics. However, on Partial Match scores, AUG_{eree} demonstrates slightly lower performance compared to TASE, and especially lower than LIQUID when employing the BERT_{base} encoder. Importantly, the performance of AUG_c consistently outperforms LIQUID and TASE on all metrics and encoders, irrespective of the encoder setting. These results demonstrate the effectiveness of our proposed framework, as well as the efficacy of the information augmentation strategy.

To be more specific, Table 1 shows comparisons of metrics among all competing models 100 backed by BERT_{base}. We can see that our proposed framework, AUG, consistently outperforms all other baselines across multispanQA dataset. Backed by BERT_{base}, AUG_c achieves EM and Overlap F1 scores of 63.05 and 79.42, respectively. Moreover, when equipped with entities' information, AUG_{eree} achieves even higher EM F1 scores of 63.93 but relatively lower Overlap F1 of 77.50 than baselines on the same encoder. These results showcase substantial improvements over the previous state-of-the-art model, LIQUID, with EM F1 score enhancements ranging from 1.61 to 2.49 percents across validation datasets.

Additionally, when utilizing RoBERTa_{base} instead of BERT_{base}, AUG_c achieves EM and Overlap F1 scores of 70.35 and 84.06 respectively on the same datasets. These scores represent EM and Overlap F1 improvements of 2.02 and 0.93 compared to the previous setup. For AUG_{eree}, the EM F1 score represents an enhancement of 0.82, with the EM and Overlap F1 values of 69.16 and 82.85 respectively. Notably, the partial metrics also indicate lower values compared to TASE and LIQUID, in line with the result supported by BERT_{base}. This is because augmenting the model with entity information, including definition and relationship knowledge, strengthens its ability to capture and understand entity concepts, which leads the model to prefer complete entity spans or empty span set as answers rather than partial entity span, and therefore a decrease in the partial recall and ultimately a lower partial F1 scores and a higher EM F1.

To further substantiate our explanation for the sub-optimal performance of our model on the Overlap F1 metric, we conducted a detailed examination of the predictions made by AUG_{eree} and two baseline models on the validation dataset. In essence, we tallied the instances where these models predicted empty answers and recalculated their Overlap F1 scores on non-empty predictions. This allowed us to investigate whether the AUG_{eree} model aligns with our hypothesis, which posits that its extensive learning of entity knowledge during training makes it inclined to output either a complete and accurate answer span or no answer at all, as opposed to a partially correct answer span.

Table 1
Approach performance on complete MultiSpanQA valid set based on BERT_{base}.

Model	Exact Match			Partial Match		
	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)
TASE	60.28	55.59	65.83	78.16	78.27	78.06
LIQUID	61.44	58.39	64.84	78.56	78.65	78.46
AUG_{c}	63.05	58.51	68.34	79.42	78.70	80.14
AUG_{eree}	63.93	60.22	68.13	77.50	77.07	77.94
AUG_{ereec}	62.90	61.02	64.89	76.72	78.56	74.97
Bagging	64.44	61.63	67.50	79.5	80.49	78.57

Table 2
Approach performance on complete MultiSpanQA valid set based on RoBERTa_{base}.

Model	Exact Match			Partial Match		
	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)
TASE	68.00	65.06	71.22	83.13	83.05	83.22
LIQUID	68.33	66.68	70.07	82.71	82.45	82.98
AUG_{c}	70.35	67.35	73.63	84.06	83.38	84.76
AUG_{eree}	69.15	67.81	70.54	82.85	83.90	81.83
AUG_{ereec}	70.44	67.83	73.26	82.80	82.45	83.15
Bagging	70.86	69.03	72.79	84.82	85.53	84.12

As presented in Table 3, AUG_{eree} indeed predicted a higher number of empty answers compared to TASE and LIQUID, while achieving relatively higher Overlap F1 scores on non-empty predictions. Specifically, when equipped with BERT_{base} as the encoder, AUG_{eree} obtained an F1 score of 0.7976 on non-empty answer predictions, whereas TASE and LIQUID scored 0.7964 and 0.7909, respectively. With RoBERTa_{base}, AUG_{eree} achieved an Overlap F1 score of 0.8414, surpassing TASE and LIQUID, which scored 0.8404 and 0.8357, respectively. Additionally, it is worth noting that AUG_{eree} consistently predicted more empty answers, whether using BERT or RoBERTa.

These findings lend support to our conjecture that the introduction of entity knowledge leads to a slight reduction in the model's Overlap F1 scores. This suggests that utilizing LLM as a knowledge source to linearize entity information from LLM text and integrate it into the original context empowers the model to acquire greater entity knowledge, thereby exhibiting a preference for more accurate and complete answer spans, or simply providing no answer.

Totally, the result, displayed in Table 2 demonstrates the same trends to Table 1. And the outcome highlights robustness in effectively generalizing across different datasets without requiring hyperparameter re-tuning.

4.3.2. Discussion

Table 1 and Table 2 discuss the performance of different augmentation integrated strategies, including the results-bagging methods whose outputs are voting results of AUG_{c}, AUG_{eree} as well as LIQUID, and the input-fusion model AUG_{ceree} who injects all kinds of information above into input contexts.

In detail, with EM f1 scores of 62.90 and 64.44 in Table 1 and 70.44 and 70.86 in Table 2, both the AUG_{ceree} and Bagging methods consistently surpass TASE and LIQUID, which exhibits robust effectiveness of information injection strategies. However, there is a little decrease caused by fusing all auxiliary information when contrast with single information augmentation strategies and the bagging method. This may be due to the likelihood that incorporating all of the augmentation information into the model inputs will confuse the model by introducing excessive auxiliary knowledge and underrepresented original context proportion. Therefore it may be more useful that adding limit information into context, and using result-bagging method, a multi-model voting to bringing all information into model with an indirect way.

From the Partial Match perspective, AUG_{ceree} and the Bagging method achieve 76.72 and 79.52 in Table 1, and 82.80 and 84.82 in Table 2. In accordance with Exact Match metrics, Bagging demonstrate a overall superior performance. Meanwhile overlap f1 score of AUG_{eree} is inferior to TASE but superior to LIQUID ,with relatively higher precision and relatively higher recall, which is comparable to all single augmented models such as AUG_{eree}. And its weak performance on overlap f1 also reveals complete entities preference of this information injection approach.

Table 3
The statistics of answers span predicted by AUG_{eree} and baselines

	BERT_{base}			RoBERTa_{base}		
	TASE(%)	LIQUID(%)	AUG_{eree}(%)	TASE(%)	LIQUID(%)	AUG_{eree}(%)
empty predictions counts	0.0245	0.0214	**0.0643**	0.0061	0.0153	**0.0291**
non-empty Overlap f1	0.7964	0.7909	**0.7976**	0.8404	0.8357	**0.8414**

Table 4
Model performance on complete MultiSpanQA valid Subset with different answer types based on BERT_{base}.

Type	**Model**	Exact Match			Partial Match		
		F(%)	P(%)	R(%)	F(%)	P(%)	R(%)
DESC	TASE	32.76	27.33	40.87	64.46	68.61	60.79
	LIQUID	36.69	31.10	44.71	66.05	65.95	66.15
	AUG_c	38.74	32.89	47.12	68.31	70.32	66.42
	AUG_{eree}	44.69	40.71	49.52	63.12	67.16	59.53
	AUG_{ereec}	40.63	38.30	43.27	64.25	71.15	58.57
	Bagging	39.91	36.69	43.75	66.70	73.68	60.93
NUM	TASE	33.33	30.23	37.14	60.98	71.99	52.89
	LIQUID	34.33	31.25	38.10	60.68	68.15	54.69
	AUG_c	35.96	33.33	39.05	64.47	71.56	58.65
	AUG_{eree}	36.20	34.48	38.10	59.02	65.63	53.62
	AUG_{ereec}	36.71	37.25	36.19	57.09	69.04	48.67
	Bagging	35.94	34.82	37.14	60.73	71.15	52.97
ENTYS	TASE	66.30	62.21	70.96	81.16	80.37	81.96
	LIQUID	67.17	65.25	69.21	81.66	81.69	81.63
	AUG_c	68.47	64.44	73.03	81.93	80.57	83.34
	AUG_{eree}	68.36	64.64	72.53	80.55	79.21	81.93
	AUG_{ereec}	67.54	65.60	69.59	79.49	80.16	78.83
	Bagging	69.65	66.94	72.59	82.31	82.07	82.55

Furthermore, we stratified the data within MultispanQA according to answer types, specifically categorizing them into DESC, NUM, and ENTYS. We subsequently conducted a comparative analysis of model performance within each of these subcategories. In particular, the results are presented in Tables 4 and Table 5, supported by BERT_{base} and RoBERTa_{base}, respectively. As indicated in Tables 4 and 5, our proposed models exhibit superior performance in terms of EM F1 scores for all categorizing. However, they demonstrate suboptimal performance in terms of overlap F1 scores.

4.3.3. Ablation Experiment

At the end of this section, we conducted ablations on our approach to confirm the effectiveness of selecting the information injection proportion. For each QA data, we randomly split the original context and the auxiliary text, then concatenated them into a final augmented context with a specific proportion to ensure that the new input length meets max_len, which has a crucial impact on our approach. In practice, we determined the final text splicing ratio by calculating the ratio of the average length of the source text to the added information, which for the AUG_c is 0.86.

Specifically, Table 6 and Table 7 displays AUG_c's performance with differential proportion to concatenate original contexts and continuation, on complete MultispanQA valid set,backed by BERT_{base} and RoBERTa_{base} respectively. We choose five proportions for information integration, which determine how much auxiliary information would be inject into each overflowed text segment. The results presented in tables indicate that using the ratio of their average lengths as the proportion of the overflow text composed of original text and auxiliary information is an effective approach. In detail, AUG_c,equipping with BERT_{base}, achieves an Exact Match F1 scores improvements of at least 4.57 compared to other proportions and an overlap F1 scores improvements of at least 2.07. In line with Table 6, when AUG_c equips with Roberta_{base}, it achieves an improvement of Exact Match F1 scores of 0.55 but an decrease of Overlap F1 scores of 0.17.

5. Conclusion

This section is not mandatory, but can be added to the manuscript if the discussion is unusually long or complex.

Table 5
Model performance on complete MultiSpanQA valid Subset with different answer types based on RoBERTa$_{base}$.

Type	Model	Exact Match			Partial Match		
		F(%)	P(%)	R(%)	F(%)	P(%)	R(%)
DESC	TASE	45.53	40.84	51.44	75.44	76.61	74.31
	LIQUID	48.68	44.76	53.37	73.27	71.57	75.04
	AUG$_c$	49.02	44.66	54.33	76.33	77.85	74.86
	AUG$_{eree}$	50.99	46.96	55.77	76.87	78.07	75.71
	AUG$_{ereec}$	49.24	44.71	54.81	71.87	73.03	70.74
	Bagging	47.70	43.78	52.40	77.43	80.63	74.47
NUM	TASE	46.23	45.79	46.67	71.89	78.42	66.36
	LIQUID	40.19	39.45	40.95	67.56	72.28	63.42
	AUG$_c$	50.69	49.11	52.38	72.86	80.22	66.74
	AUG$_{eree}$	42.40	41.07	43.81	66.67	73.51	61.00
	AUG$_{ereec}$	45.58	44.55	46.67	65.86	73.19	59.87
	Bagging	44.65	43.64	45.71	70.11	78.49	63.35
ENTYS	TASE	72.57	69.94	75.41	84.90	84.32	85.48
	LIQUID	72.95	71.77	74.16	85.02	84.75	85.29
	AUG$_c$	74.59	71.87	77.53	85.79	84.40	87.23
	AUG$_{eree}$	73.50	72.81	74.22	84.74	85.49	83.99
	AUG$_{ereec}$	75.04	72.81	77.41	85.37	84.46	86.29
	Bagging	75.85	74.52	77.22	86.73	86.73	86.74

Table 6
Ablations of AUG$_c$ on different proportion for information concatenation, based on BERT$_{base}$.

Proportion	Exact Match			Partial Match		
	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)
0.90	58.55	57.69	59.44	71.88	74.30	69.61
0.70	61.70	59.03	64.63	76.58	77.84	75.36
0.50	61.22	57.37	65.62	77.38	77.61	77.16
0.40	61.36	56.50	67.13	78.26	77.57	78.96
0.10	61.41	56.36	67.45	78.94	78.45	79.44
0.14	63.05	58.51	68.34	79.42	78.70	80.14

Acknowledgements

This research was supported by the Natural Science Foundation of Zhejiang Province, China (Grant No. LQ22F020027), Fundamental Research Funds of Zhejiang Sci-Tech University (Grant No. 23232138-Y), Liaoning Provincial Natural Science Foundation of China (Grant No. 2022-KF-21-01) and the Key Research and Development Program of Zhejiang Province, China (Grant No. 2023C01041 and 2022C01079).

Table 7
Ablations of AUG$_c$ on different proportion for information concatenation, based on RoBERTa$_{base}$.

Proportion	Exact Match			Partial Match		
	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)
0.90	58.13	56.92	59.39	71.70	73.49	69.98
0.70	66.97	65.42	68.60	80.05	80.80	79.32
0.50	68.94	67.23	70.75	82.43	83.23	81.64
0.40	68.95	66.11	71.06	83.71	83.71	81.71
0.10	69.80	66.47	73.47	84.23	83.49	84.99
0.86	70.35	67.35	73.63	84.06	83.38	84.76