Information Processing and Management 61 (2024) 103678

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

A Token-based transition-aware joint framework for multi-span question answering

Zhiyi Luo, Yingying Zhang, Shuyun Luo *

School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China

ARTICLE INFO

Keywords:
Reading comprehension
Multi-span question answering
Pretrained language models
Multitask learning
Chinese datasets

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

Extractive question answering, also commonly referred as the task of reading comprehension, which aims to answer a user's question by finding short text segments (i.e., *spans*) from the given context, has been actively studied and achieved rapid progress in recent years. Benefiting from the sophisticated industrial search engines and the vast amount of text collections on the web, high-quality contexts relevant to the user questions can be effectively retrieved to construct the datasets. As a result, the answers directly drawn from the high-quality contexts are expressive enough. Hence, existing datasets and models (Dasigi, Liu, Marasovic, Smith, & Gardner, 2019; Lee, Kim, & Kang, 2023; Li, Tomko, Vasardani, & Baldwin, 2022; Liu, Mao, Geng, & Cambria, 2023), cast reading comprehension as an *extractive* task that is easy to learn.

Most previous work (Rajpurkar, Jia, & Liang, 2018; Rajpurkar, Zhang, Lopyrev, & Liang, 2016; Yang et al., 2018; Zaheer et al., 2020) restricts the answer extracted from the context to a single text span which can only satisfy limited real-world open-domain questions. For example, as shown in Table 1 (the left example), the complete answer to a question could consist of a series of non-contiguous spans, or even the question itself could have multiple intents, where the answer to each intent is composed of one or more spans drawn from the input. Hence, the models of multi-span question answering have significant utility to users.

* Corresponding author.
E-mail addresses: luozhiyi@zstu.edu.cn (Z. Luo), 272831920@qq.com (Y. Zhang), shuyunluo@zstu.edu.cn (S. Luo).
URL: http://zhiyiluo.site (Z. Luo).

https://doi.org/10.1016/j.ipm.2024.103678
Received 9 July 2023; Received in revised form 10 December 2023; Accepted 24 January 2024

Table 1
Examples of questions and answers.

Dataset	CLEAN	MultiSpanQA
Question	如何养好约克夏犬? (How to raise a Yorkshire terrier?)	Who wrote the song if you could see me now?
Context	想要养好约克夏对于它们的饮食护理都要做好，由于它们所需的运动量并不是特别大，所以一般在家里就可以，不过还是要适当地带它们出去运动一下，这样它们能够生长的更好。 (... must do a good job in their diet and care. ... but it is still necessary to take them out for some exercise so that they can grow better.)	"If You Could See Me Now" is a 1946 jazz standard, composed by Tadd Dameron. He wrote it especially for vocalist Sarah Vaughan, a frequent collaborator. Lyrics were written by Carl Sigman and it became one of her signature songs, inducted into the Grammy Hall of Fame in 1998.
Answer	Segment1: 饮食护理 (diet and care) Segment2: 适当地带它们出去运动一下 (take them out for some exercise)	Segment1: Tadd Dameron Segment2: Carl Sigman

Recently, Segal, Efrat, Shoham, Globerson, and Berant (2020) cast the multi-span extraction as a sequence tagging task, predicting whether each token is part of an answer. Li et al. (2022) captured the global information by integrating two sub-tasks of predicting the number of spans to extract and the answer structure annotated in their proposed dataset. Benefiting from nowadays pretrained models (e.g., BERT Kenton & Toutanova, 2019, RoBERTa Liu et al., 2019), the above neural approaches have achieved promising performance on answer span extraction, especially for factoid questions where the expected answers are *entities* (e.g., *Person* and *Location*).

However, existing systems fail to carefully consider span boundaries according to the information need of the question, and thus have very limited capabilities of precisely drawing a *description* answer, for example, the answer spans of "饮食护理 (diet and care)" and "适当地带它们出去运动一下 (take them out for some exercises)" as shown in Table 1, from the input. Moreover, in the construction of existing datasets, user question candidates are constrained by the organization way of the document collection where the relevant contexts are retrieved from. For example, Wikipedia, the most widely used document collection for context retrieval, organizes articles using entities. The problem is that a *semantic gap* exists between the real context of what users want to ask and the entity-based articles (i.e. single Wiki page). Backed by the document collection of Wikipedia, existing datasets include most types of the factoid questions, whereas they exclude many questions that cannot be answered from a single article of an entity.

Following the above observations, we in this paper propose a novel approach to explicitly model the implicit *neighboring transitions* via the adjacent word (or token) relations, which are both semantically and syntactically informative for span boundaries identification. This approach captures the intuition that the span boundary is produced in-between adjacent words. We indicate neighboring transitions in-between as five types of relations. Then, we jointly learn the sequence tagging task together with the adjacent word relation classification task evolved from the span boundary identification, considering whether each token is part of an answer and which span each token belongs to accordingly. With the awareness of adjacent word relations, we incorporate the information of span boundaries in a multi-task learning manner. Furthermore, in order to ameliorate the observed limitation of previous datasets, which including more real-world open-domain questions, we create a new dataset (in Chinese) in a more natural manner, extracting question–context pairs from a large-scale knowledge Q&A sharing platform (i.e., BaiduZhidao[1]) instead of Wikipedia.

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

- We create a new reading comprehension dataset named CLEAN[2] that consists of both single-span and multi-span answers, covering a wide range of open-domain question topics. CLEAN overcomes the constraints imposed by previous datasets by incorporating carefully crafted long answers as contexts, effectively bridging the semantic gap with the insights provided by respondents.
- We propose a novel approach for multi-span reading comprehension, where we explore implicit neighboring transitions using adjacent word relations, effectively capturing both semantic and syntactic information pertaining to the boundaries of answer spans.
- We demonstrate that incorporating the span boundary information via the awareness of adjacent word relations improves strong baselines on three multi-span question answering datasets (both English and Chinese).

2. Preliminary

In this section, we present the problem statement and then introduce the datasets.

[1] https://zhidao.baidu.com/.

[2] We intend to make CLEAN 1.0 dataset publicly avaiable at http://zhiyiluo.site/misc/clean_v1.0_sample.json for future work in this research area.

2.1. Problem statement

We formulate the problem of multi-span question answering as a task of multi-span extraction on the basis of the reading comprehension (RC) datasets. The objective of the task is to extract one or more answer spans based on the input question and context.

Formally, given an input question represented as a sequence of words $q = (q_1, q_2, \ldots, q_n) \in \mathcal{V}$, and an input context c, associating to q, which is also represented as a sequence of words $c = (c_1, c_2, \ldots, c_m) \in \mathcal{V}$, where $\mathcal{V}$ refers to the vocabulary, the objective of the system is to extract one or more spans as the answer a, say l spans, $a = [a_1, a_2, \ldots, a_l]$ from the context c, where the ith span a_i is a sequence of words $a_i = (c_{i_s}, c_{i_s+1}, \ldots, c_{i_e})$, ranging from the start position i_s to its end position i_e. Note that the extracted answer spans should be neither duplicated nor overlap with each other. That means, for any i, j where $i < j$, $i_e < j_s$ holds.

Evaluation are exact match and partial match with any of the reference answer strings after minor normalization such as lowercasing, following evaluation scripts from Li et al. (2022).

2.2. Datasets

In this work, we conduct the experiments on three RC datasets, ranging from two English datasets and one proposed Chinese dataset for multi-span question answering.

The latest dataset, MultiSpanQA, focus on multi-span questions, which is derived from Natural Question (NQ) (Kwiatkowski et al., 2019), a large-scale open domain QA dataset. It also has an expanded variant by introducing single-span and unanswerable questions, namely MultiSpanQA (expand).

In MultiSpanQA and MultiSpanQA (expand), the questions are derived from real queries issued to the Google search engine. Each question is paired with a context extracted from a retrieved Wikipedia page. However, it is important to acknowledge that Wikipedia pages are structured around entities, which may not fully align with the intentions of real-world open-domain questions, especially those that are non-factoid in nature. This entity-centric organization of Wikipedia pages introduces a semantic gap between the content and the intended meaning of real-world open-domain questions. As a result, existing RC datasets like MultiSpanQA are limited to questions that primarily revolve around entities, allowing for the extraction of concise spans from a single Wikipedia page as answers. Moreover, there is a noticeable scarcity of publicly available open-domain multi-span QA datasets in the Chinese language. Although recently proposed CMQA (Ju et al., 2022) is one such dataset, it primarily focuses on a new task of conditional question answering, which may not directly address the requirements of traditional question answering scenarios.

To ameliorate these limitations, we create a more realistic and challenging dataset, utilizing a large-scale Chinese online knowledge Q&A sharing platform (i.e. BaiduZhidao) which is full of open-domain questions with crafted long answers from public users. Over the course of two decades of crowd-sourcing efforts, the shared questions have covered a broad range of subjects, including people, celestial bodies, flora and fauna, landmarks, etc. Firstly, we conduct a random crawl of one million questions across 29 popular subjects in the open domain. Next, we employ two key strategies to increase the proportion of multi-span answers relative to single-span answers: (1) selectively choosing questions that contain keywords such as 如何/怎样 (e.g. *how*) or 哪些 (e.g. *what/which ones*), as these types of questions typically require multipart descriptions; and (2) translating all the crawled questions into English and specifically targeting questions featuring plural nouns. Note that, a question may have a number of long answers created by different users, and we only consider those with more than 3 likes as potential candidates. Then, our annotators are educated to pick up high-quality contexts from the candidate answers for each question and identify one or more answer spans within the context that can effectively answer the given question. Finally, we obtain a diverse collection of multi-span answers in various formats.

In our proposed **C**hinese mu**L**ti-span qu**E**stion **AN**swering (or CLEAN) dataset, the context consists of meticulously constructed long answers that effectively bridge the semantic gap with the insights provided by respondents. This approach ensures that the question intents are appropriately addressed and breaks the constraints imposed by previous datasets in terms of question selection. Table 1 (left) presents a specific example from the CLEAN dataset, where the question is obtained from BaiduZhidao, and the context is selected from the original long answers associated with that question. To make our dataset more general, as well as considering the versatility of questions, unlike MultiSpanQA, we only annotate the span of answer without the structure of the answer.

3. Our framework

In this section, we first introduce a basic neural framework for multi-span question answering based on pretrained language models, which involves casting multi-span extraction as a sequence tagging task. Intuitively, answer span boundaries often align with semantic and syntactic shifts within contexts, the identification of which is more subtle in multi-span extraction (see the example in Table 2). We then propose a novel joint framework called **TO**ken-b**AS**ed **T**ransition-aware (TOAST) for multi-span question answering. TOAST effectively captures these shifts through token-based transition awareness and incorporates boundary knowledge into token representations to enhance tagging predictions.

The architecture of TOAST is depicted in Fig. 1. The *Transition Identification* module demonstrates how span boundary identification can be transferred to a token-based transition classification task, while the *Transition Incorporation* module explains how we incorporate semantic and syntactic shifts in adjacent word relations to improve the informativeness of the extraction. In addition to the sequence tagging task (Task 1 in Fig. 1), TOAST also includes an auxiliary task of transition classification (Task 2 in Fig. 1). We therefore present a multi-task learning strategy for TOAST. Finally, we present the joint decoding step, which leverages outputs from multiple tasks to improve performance.

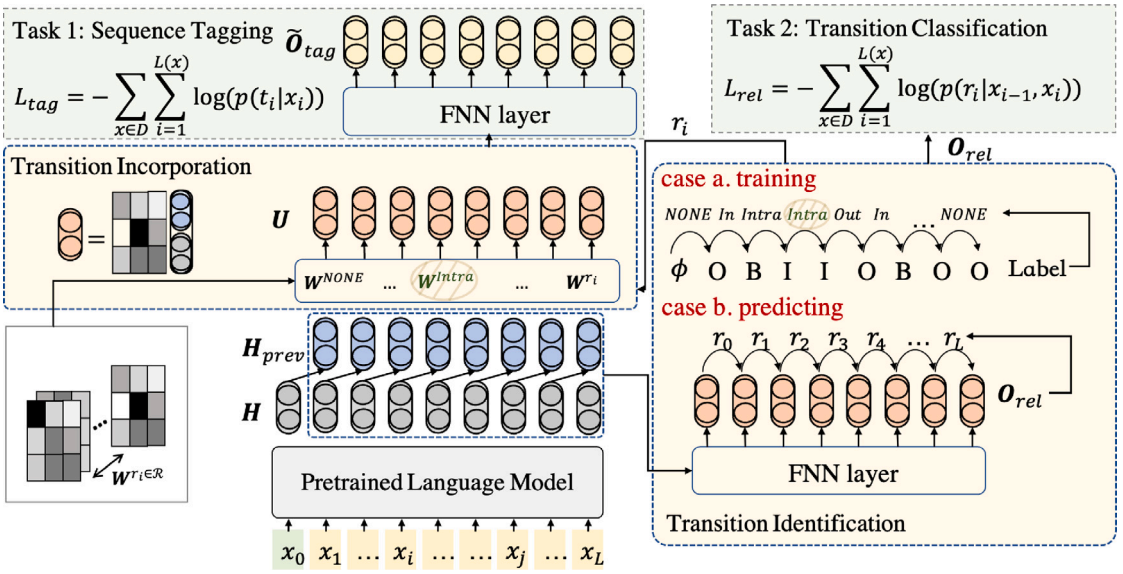


Fig. 1. An illustration of TOAST framework architecture.

3.1. A BERT-based sequence tagging model

Extracting a variable number of spans from an input text can be commonly cast as a sequence tagging problem, which is suitable for answer extraction of multi-span questions. Following the observation of Li et al. (2022), we adopt the BIO tagging scheme to mark answer spans in the context where words are tagged as either part of the answer (**B**egin, **I**nside) or not (**O**ther). Formally, BIO tagging scheme is represented by a tag set $\mathcal{T}$ ={ B, I, O}.

3.1.1. Encoder

First, we encode the question and context with a pretrained language model, such as BERT and RoBERTa. Generally, the encoder takes in the input text and encodes it into a series of contextualized hidden representations. Specifically, the BERT-style encoder takes in a $[CLS]$ token, followed by the concatenation of a question $q = (q_1, q_2, \dots, q_n)$ and its context $c = (c_1, c_2, \dots, c_m)$ with $[SEP]$ token, as the input sequence $([CLS], q_1, q_2, \dots, q_n, [SEP], c_1, c_2, \dots, c_m)$. We use $x = (x_1, x_2, \dots, x_L)$ as a more concise notation of the above sequence, where L refers to the length of the input, that $L = n+m+2$ when ignoring padding tokens. Then we send it through neural layers of the encoder to finally obtain hidden representations $\mathbf{H} = \left[\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_L\right] \in \mathbb{R}^{L\times d_h}$, where d_h denotes the hidden-layer size, and $\mathbf{h}_i$ represents the hidden representation of the ith input token. In the next step, output representations are sent into an FFN (Feed Forward Neural Network) layer, and computed the output logits as $\mathbf{O}_{tag} = FFN(\mathbf{H}) \in \mathbb{R}^{L\times|\mathcal{T}|}$ for token-level tag prediction, where $\mathcal{T}$ denotes the tag set we described above.

3.1.2. Training & Decoding

The task of token-level tag prediction that aims to assign a specific tag t_i from the set $\mathcal{T}$ to each token x_i in the input sequence x, is inherently a classification problem. In this regard, we employ the widely adopted cross-entropy loss, which aggregates the negative logarithm of the predicted probability for the correct tag across all tokens in the training set. This loss penalizes the model heavily for confident yet incorrect predictions, prompting the model to adjust its parameters and minimize the loss. To optimize the answer span tagging model, we compute the cross-entropy loss, say L_{tag}, as follows:

$$L_{tag} = -\sum_{x\in D}\sum_{i=1}^{L(x)} \log(p(t_i|x_i)), \tag{1}$$

where D is the training set, and $L(x)$ is the input length of x. At the decoding step, we predict the tag $\hat{t}_i$ for token x_i as follows:

$$\hat{t}_i = \underset{t\in\mathcal{T}}{\operatorname{argmax}}\, p(t|x_i). \tag{2}$$

Then, we extract a successive sequence of B- and I-tagged tokens as an answer span. For example, the input sequence of context “a b c d e f” tagged as “O B I O B O” can be decoded into two answer spans of “b c” and “e” respectively. Note that we do a post-processing step to tag “O” for all input tokens except for those from the context, which means we only draw answer spans from the input context.

Table 2
The definition of adjacent word relations.

Tag Pair (t_{i-1}, t_i)	Span Transition	Relation Name	MSQA Example
(O, O)	NONE	NONE	Question : When are the Winter Olympics and where are they ?
(O, B)	Other → Span1	In relation	… event held **between 9 and 25 February**
(B, I), (I, I)	Span1 → Span1	Intra relation	Labeled Context: **2018 in Pyeongchang County , Gangwon**
(B, B), (I, B)	Span1 → Span2	Inter relation	**Province , South Korea** , with the opening …
(B, O), (I, O)	Span1 → Other	Out relation	Gold Spans: ["between 9 and 25 February 2018", "in Pyeongchang County , Gangwon Province , South Korea"]

3.2. Token-based transition awareness

One limitation of the baseline framework is that it often breaks a complete answer span into many meaningless splits, which is especially serious for those description answer spans. This reflects that the model lacks specific knowledge to indicate span boundaries accurately. A context or long answer for a multi-span question contains a variable number of answer spans corresponding to one or more user intents. Therefore, semantic meanings and syntactic patterns transit quickly across words (or tokens), the minimal text unit. Following the above observation, we propose a novel approach to capture such token-based neighboring transitions, then come up with an auxiliary task and joint learning strategy to incorporate such knowledge into our framework.

3.2.1. Transition identification

In our approach, neighboring transitions are represented as five types of *relations* in-between adjacent words by exploring whether a span transition happens across the adjacent words. More specifically, for each adjacent word pair in the input, say x_{i-1} and x_i, 8 possible tagging cases exist under the BIO tagging scheme, denoted as (t_{i-1}, t_i), where $t_i \in \mathcal{T}$ is the ground truth tag of x_i. As shown in Table 2, we summarize five relations in-between adjacent words (x_{i-1}, x_i) according to their tags (t_{i-1}, t_i), each of which corresponds to a type of span transition. For example, an inter-span transition (defined as *Inter* relation) happens when an adjacent word pair is tagged as (B, B) or (I, B), which means x_{i-1} and x_i are both part of the answer while belong to two consecutive but different answer spans. The relation of *NONE* indicates that there is no span transition in-between x_{i-1} and x_i. Note that x_i could be either a context word (e.g., c_i) or a question word (e.g., q_i) as well as a special token (e.g., [CLS] and [SEP]). The NONE relation holds if any word in (x_{i-1}, x_i) is a non-context word or both x_{i-1} and x_i are context words, but neither is part of the answer. To further illustrate the types of relations, we provide an example from the MSQA dataset in Table 2. The question "When are the Winter Olympics and where are they" has two intents: querying the time and location of the Winter Olympics. The *Inter* relation between the terms "2018" and "in" indicates a semantic transition from time to location. By exploring every neighboring transition modeled by the defined set of relations between adjacent words, our approach effectively captures semantically and syntactically span-based transitions. Next, we present how to encode span-transition knowledge via adjacent word relations in Table 2 and incorporate such knowledge into our framework.

3.2.2. Transition incorporation

After transition identification, each adjacent word pair (x_{i-1}, x_i) has been associated with a relation $r_i \in \mathcal{R}$, where $\mathcal{R}$ is the set of adjacent word relations. As defined in Table 2, $\mathcal{R} = \{NONE, In, Intra, Inter, Out\}$. Then, we equip our model with token-based transition knowledge though relational transformations. More specifically, for each (x_{i-1}, r_i, x_i) triple, that r_i associates to a relational matrix $\mathbf{W}^{r_i} \in \mathbb{R}^{d_h \times 2d_h}$, we compute the enhanced contextualized representation of x_i with the awareness of span transitions as follows:

$$\mathbf{u_i} = \mathbf{W}^{r_i} \left[\mathbf{h}_{i-1}; \mathbf{h}_i\right], \tag{3}$$

where $\mathbf{h_i}$ is the output representation of token x_i from the encoder described in Section 3.1, $\mathbf{u_i}$ is its enhanced representation, and $[;]$ is vector concatenation across row. Then, we send the enhanced representations $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_L] \in \mathbb{R}^{L \times d_h}$ into a FNN layer and compute the output logits as:

$$\widetilde{\mathbf{O}}_{tag} = FNN(\mathbf{U}) \in \mathbb{R}^{L \times |\mathcal{T}|}, \tag{4}$$

where $[,]$ is vector concatenation across column, and the implicit multiplication is matrix multiplication. We make up a fake token x_0 as the prior word of x_1, that $(x_0, NONE, x_1)$ holds, and $\mathbf{h}_0$ is set to be a vector full of zeros. Hence, each input token x_i can be paired with its prior token x_{i-1} as an adjacent word pair indicating with a relation r_i, where i ranges from 1 to L.

We argue that the awareness of span-based transition which intuitively modeled by adjacent word relations bring extra information for span identification, thus facilitate the multi-span extraction task. The proposed incorporation method is natural and straightforward, and we will demonstrate its effectiveness in our experiments (See Section 4).

3.3. Multi-task learning

To aware the knowledge of span-based transition, we need to construct a series of relations $(r_1, r_2, \dots, r_L)$ from the input sequence $(x_1, x_2, \dots, x_L)$, where r_i indicates the adjacent word relation in-between the adjacent word pair (x_{i-1}, x_i). For the training data, each input token x_i is annotated with a ground truth tag t_i, thus we can draw relation r_i from the tag pair (t_{i-1}, t_i) according to Table 2. While it is not true for the test data, we cannot harvest accurate adjacent word relations directly. Thus, on the basis of previous sequence tagging model, we build a multi-task learning framework by introducing an auxiliary task of token-based transition classification. This relation classifier of the auxiliary task shares the BERT-style encoder with previous tagging model with a new, unshared FNN layer upon. The output logits of relation classifier are computed as:

$$\mathbf{O}_{rel} = FNN([\mathbf{H}_{prev}; \mathbf{H}]) \in \mathbb{R}^{L \times |\mathcal{R}|}, \tag{5}$$

where $\mathbf{H}_{prev} = [\mathbf{h}_0, \mathbf{h}_1, \dots, \mathbf{h}_{L-1}] \in \mathbb{R}^{L \times d_h}$ denotes representations of the prior token sequence $(x_0, x_1, \dots, x_{L-1})$ with respect to the input sequence $(x_1, x_2, \dots, x_L)$. For the auxiliary task, we use cross-entropy loss to train the relation classifier, which reflects the awareness of span-based transition. More specifically, the above loss L_{rel} is computed as:

$$L_{rel} = -\sum_{x \in D} \sum_{i=1}^{L(x)} \log(p(r_i | x_{i-1}, x_i)), \tag{6}$$

where D refers to the training data, and $L(x)$ denotes the length of input sequence x. To jointly learn shared parameters in the encoder as well as leverage the representation of token-based transition knowledge, we use a combinartorial loss as follows:

$$L = L_{tag} + L_{rel}. \tag{7}$$

In summary, the strategy of our multi-task learning framework is that, given an input sequence, we first predict the token-based transitions in each position, then leverage the predicted transitions to calculate the enhanced token representation following Eq. (4), finally optimize the proposed combinartorial loss in Eq. (7) to update joint model parameters together.

3.4. Decoding

Next, we present two decoding strategies for our joint framework. As described above, we jointly train two tasks (i. e, sequence tagging and transition classification) in our framework, then predict the adjacent token-based transitions and the token tags respectively. Intuitively, we can decode the answer spans from tags predicted by the enhanced sequence tagging model of our framework directly, similar to the decoding algorithm described in Section 3.1. Formally, we decode tag $\hat{t}_i$ of x_i as follows:

$$\hat{t}_i = \underset{t \in \mathcal{T}}{\operatorname{argmax}}\, p(t | x_i, \mathcal{M}_{tag}), \tag{8}$$

where $\mathcal{M}_{tag}$ represents the token-based transition aware tagging model (see Section 3.2). As we will see in Section 4, this enhanced tagging model outperforms the baseline model marginally.

While the predicted tags implicitly incorporate the knowledge of token-based transitions, we propose to explicitly combine the predictions from the enhanced tagging model $\mathcal{M}_{tag}$ with those from the transition classifier $\mathcal{M}_{rel}$ at decoding time. We score the potential tag t of a token x_i, considering not only x_i but also its prior token x_{i-1} through the predicted transition r_i in-between.

To be more specific, we jointly decode the tag t_i of x_i as follows:

$$\hat{t}_i = \underset{t \in \mathcal{T}}{\operatorname{argmax}}\, \widetilde{S}(t | x_{i-1}, x_i, \hat{t}_{i-1}, \mathcal{M}_{joint}), \tag{9}$$

where $\widetilde{S}$ denotes the scoring function that intakes the predicted transition between x_{i-1} and x_i. Therefore, $\widetilde{S}(t | x_{i-1}, x_i, \hat{t}_{i-1}, \mathcal{M}_{joint})$ is computed as:

$$\begin{aligned}
\widetilde{S}(t | x_{i-1}, x_i, \hat{t}_{i-1}, \mathcal{M}_{joint}) &= S(t | x_i, \mathcal{M}_{tag}) + S(\hat{r}_i | x_{i-1}, x_i, \hat{t}_{i-1}, t, \mathcal{M}_{rel}) \\
&= S(t | x_i, \mathcal{M}_{tag}) + \mathbb{1}(\hat{t}_{i-1} \in \text{First}(\hat{r}_i)) S(\hat{r}_i | x_i, x_{i-1}, \mathcal{M}_{rel})^{\frac{1}{2}} \cdot \mathbb{1}(t \in \text{Second}(\hat{r}_i)) S(\hat{r}_i | x_i, x_{i-1}, \mathcal{M}_{rel})^{\frac{1}{2}} \\
&= S(t | x_i, \mathcal{M}_{tag}) + \mathbb{1}(\hat{t}_{i-1} \in \text{First}(\hat{r}_i)) \cdot \mathbb{1}(t \in \text{Second}(\hat{r}_i)) \cdot S(\hat{r}_i | x_i, x_{i-1}, \mathcal{M}_{rel}),
\end{aligned} \tag{10}$$

where S refers to the probability score, $\mathbb{1}$ denotes the indicator function, First(r) represents a set of tags consisting of the distinct first elements of all tag pairs mapped from r according to Table 2, and Second operates like First except that Second collects the second elements. For example, *Inter* can be mapped into two kinds of tag pairs (B, B) and (B, I), then First$(Inter)$ is the set {B} and Second$(Inter)$ is the set {B, I}. Note that, $\hat{r}_i$ in Eq. (10) is the optimal transition relation predicted by $\mathcal{M}_{tag}$ which is computed as:

$$\hat{r}_i = \underset{r_i \in \mathcal{R}}{\operatorname{argmax}}\, p(r_i | x_{i-1}, x_i, \mathcal{M}_{rel}). \tag{11}$$

Table 3
Dataset statistics.

Dataset	Language	#Examples	#Multi-span examples
MSQA (Li et al., 2022)	English	6536	6536
MSQAExp (Li et al., 2022)	English	19,608	6536
CLEAN (ours)	Chinese	9063	4204

Table 4
Proportion and examples of answer types in MSQA and CLEAN datasets.

Dataset	Answer Type	%	Example
MSQA	Description	16.40	*other gases*
	Entity	76.30	*Vermont*
	Numeric	7.30	*9,677 ft*
CLEAN	Description	76.54	长满了尖锐的刺 *(full of sharp thorns)*
	Entity	18.00	北京市 *(Beijing)*
	Numeric	5.46	*1006 万人 (10.06 million people)*

4. Experiments

In this section, we compare TOAST with multiple strong baselines on multi-span question answering. We first introduce the datasets and experimental setup, then show the experimental results and analysis for different models.

4.1. Dataset description

We conduct the experiments on three RC datasets, including two English datasets, namely MultiSpanQA and its variant MultiSpanQA (expand), as well as one Chinese dataset, namely CLEAN (ours). Table 3 shows an overview statistics of those datasets. To examine the performance of different models on various question types, we categorize the samples into two types: *description* and *entity*, based on the expected answer type. Table 4 illustrates the breakdown of these types along with an example for each answer type class. Note that, MSQA is short for MultiSpanQA, and MSQA-Exp is short for the expansion of MultiSpanQA.

MultiSpanQA

MultiSpanQA (Li et al., 2022), a recently proposed dataset for multi-span question answering, consists of 6.5 K multi-span examples, where the questions are user queries issued to Google search engine and the contexts are extracted from English Wikipedia.

MultiSpanQA (expand)

MultiSpanQA(expand) (Li et al., 2022), an expanded variant of MultiSpanQA, intakes single-span and unanswerable questions, and consists of 19 K examples in total.

CLEAN

We propose CLEAN harvesting from a Chinese online knowledge Q&A sharing platform (i.e., BaiduZhidao), which consists of 9063 examples in total, including over 4.2K multi-span examples. CLEAN contains 3077 distinct open-domain user questions, and each question is annotated with a number of contexts from its associated long answers. To be more specific, we recruit three annotators to extract answer spans from given contexts, and retained only those examples that achieved a Fleiss' Kappa score greater than 0.5. Our results indicate a high level of inner-annotator agreement, with a Fleiss' Kappa score of 0.739, suggesting that the annotations are consistent and reliable.

4.2. Experimental setup

For all competing models and our models, we use the HuggingFace implementation of BERT_{base} or RoBERTa_{base} as the *encoder* with $max\_len = 512$. Specifically, for TOAST models, we initialize the learning rate to $3e{-}5$ and set the batch size to 32, then 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 using the model checkpoints obtained after 50 epochs. As for the other competing models, we follow the configurations as originally reported in Lee et al. (2023), Li et al. (2022) and Segal et al. (2020) accordingly. Next, we introduce the comparison models and the evaluation metrics in our experiments.

4.2.1. Models under comparison

We introduce four competing models for multi-span answer extraction. These are **SSE** (Devlin, Chang, Lee, & Toutanova, 2019), **TASE** (Segal et al., 2020), **SNP-TASE** (Li et al., 2022) and **LIQUID** (Lee et al., 2023).

SSE is a traditional single-span extraction model which formulates the answer extraction task by span labeling, i.e., identifying in the context a span (a continuous string of text) that constitutes an answer. SSE builds a learnable linear layer upon the encoder which is used to predict the start and end position of the span. We adapt SSE for multi-span question answering, that considers the first span of a multi-span answer as the ground truth at the training step.

TASE is a tag-based span extraction model which identifies the multi-span answer by assigning a tag to every input token with BIO tagging scheme. Following Li et al. (2022), we build a strong baseline upon TASE, integrating with span number prediction and structure prediction, namely **SNP-TASE**.

The most recent framework, **LIQUID** (Lee et al., 2023), employs question generation as a form of data augmentation to enhance answer extraction performance, differing from the conventional emphasis on the exploration of model architectures.

According to the different strategies of decoding, we propose two models, $\textbf{TOAST}_{tag}$ and $\textbf{TOAST}_{joint}$, on the basis of our framework, where **TOAST** is our joint learning framework, and the respective suffix designates the module from which predictions are used at decoding step. $\textbf{TOAST}_{tag}$ predicts a tag for every token using the predictions of $\mathcal{M}_{tag}$ (see Eq. (8)), while $\textbf{TOAST}_{joint}$ leverages predictions from both $\mathcal{M}_{tag}$ and $\mathcal{M}_{rel}$ explicitly (see Eq. (9)).

4.2.2. Evaluation metrics

We evaluate the performance of our methods by automatic metrics and human evaluation.

Automatic Metrics. We use two automatic metrics for evaluation: Exact Match and 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.

The exact match shows the quality of predictions straightforwardly. However, counting the number of exact matches makes the score discrete and coarse. Besides, it penalizes the partial matched prediction too much. The overlap F1 score considers the overlap between the prediction and gold and thus can be used as a complementary metric to the exact match metric.

Human Evaluation. We randomly select 50 samples from each dataset and average the scores of two human annotators who are proficient in English and Chinese. We ask the annotators to score each sample under 3 span-level aspects (completeness, correctness and distinctness) and the overall quality:

- **Completeness** is to measure the semantic completeness of a given predicted span. If the span expresses the meaning completely, it will receive a high rate on the completeness.
- **Correctness** is to measure the content correctness of a given predicted span. If the span answers the user question correctly, it will receive a high rate on the correctness.
- **Distinctness** is to measure the content distinctness of a given predicted span against other predictions. If the span distinguishes from others semantically, it will receive a high rate on the distinctness.
- **Overall** is to measure the overall quality of the predictions from a model for each sample.

For a given sample, each aspect is judged as a five-scale scores, ranging from 1 (Poor) to 5 (Excellent) depending on its quality. For each span-level aspect, we score every predicted span and then average the scores across all predicated spans for evaluation. In addition, for each sample we use the averaged overall scores from three annotators to present the overall quality of the predictions of that sample.

4.3. Experimental results and analysis

In this section, we compare TOAST with all competing models described above both quantitatively and qualitatively.

4.3.1. Comparison results

We evaluate our model as well as baselines (Section 4.2.1) on the development splits of four datasets (Section 4.1) using both automatic metrics and human evaluation metrics (Section 4.2.2). The comparison results are shown in Table 5 (exact match), Table 6 (overlap) and Fig. 2 (human evaluation).

Tables 5 and 6 illustrate the performance comparison between the proposed models, TOAST_{tag} and TOAST_{joint}, 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. Notably, TOAST_{tag} exhibits superior performance in EM F1 across all three datasets. However, when employing the BERT_{base} encoder on the MSQA dataset, TOAST_{tag} demonstrates slightly lower performance in overlapped F1 compared to LIQUID. Importantly, the performance of TOAST_{joint} consistently outperforms LIQUID on all three datasets, irrespective of the encoder setting. These results demonstrate the effectiveness of our proposed framework, as well as the of the joint decoding strategy.

To be more specific, Table 5 shows the comparisons of exact match scores among all competing models. We can see that our proposed framework, TOAST, consistently outperforms all other baselines across multiple multi-span question answering datasets.

Table 5
The exact match results for all competing models, including micro-average precision (P), recall (R) and f1 score (F1).

	MSQA			MSQA-Exp			CLEAN		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
[$BERT_{base}$]									
SSE	52.53	17.95	26.76	32.21	19.61	24.38	22.02	21.19	21.60
TASE	54.16	63.63	58.52	51.01	59.12	54.77	28.73	49.39	36.33
SNP-TASE	58.12	60.50	59.28	58.32	60.34	59.31	–	–	–
LIQUID	55.67	66.24	60.50	55.56	61.49	58.37	41.04	47.95	44.23
$TOAST_{tag}$	60.10	66.61	63.19	59.47	61.98	60.70	57.95	62.31	60.05
$TOAST_{joint}$	**60.55**	**66.82**	**63.53**	**60.97**	**65.57**	**63.19**	**58.48**	**62.64**	**60.49**
[$RoBERTa_{base}$]									
SSE	55.13	18.84	28.08	33.18	20.20	25.12	23.41	22.74	23.07
TASE	63.91	69.13	66.42	64.17	68.20	66.12	33.62	51.05	40.54
SNP-TASE	61.43	67.30	64.23	32.85	22.41	26.64	–	–	–
LIQUID	66.14	72.16	69.02	63.80	66.21	64.98	47.75	56.20	51.63
$TOAST_{tag}$	68.15	73.57	70.76	66.90	68.23	67.56	61.04	64.32	62.64
$TOAST_{joint}$	**68.61**	**73.89**	**71.15**	**68.41**	**69.01**	**68.71**	**61.86**	**64.52**	**63.16**

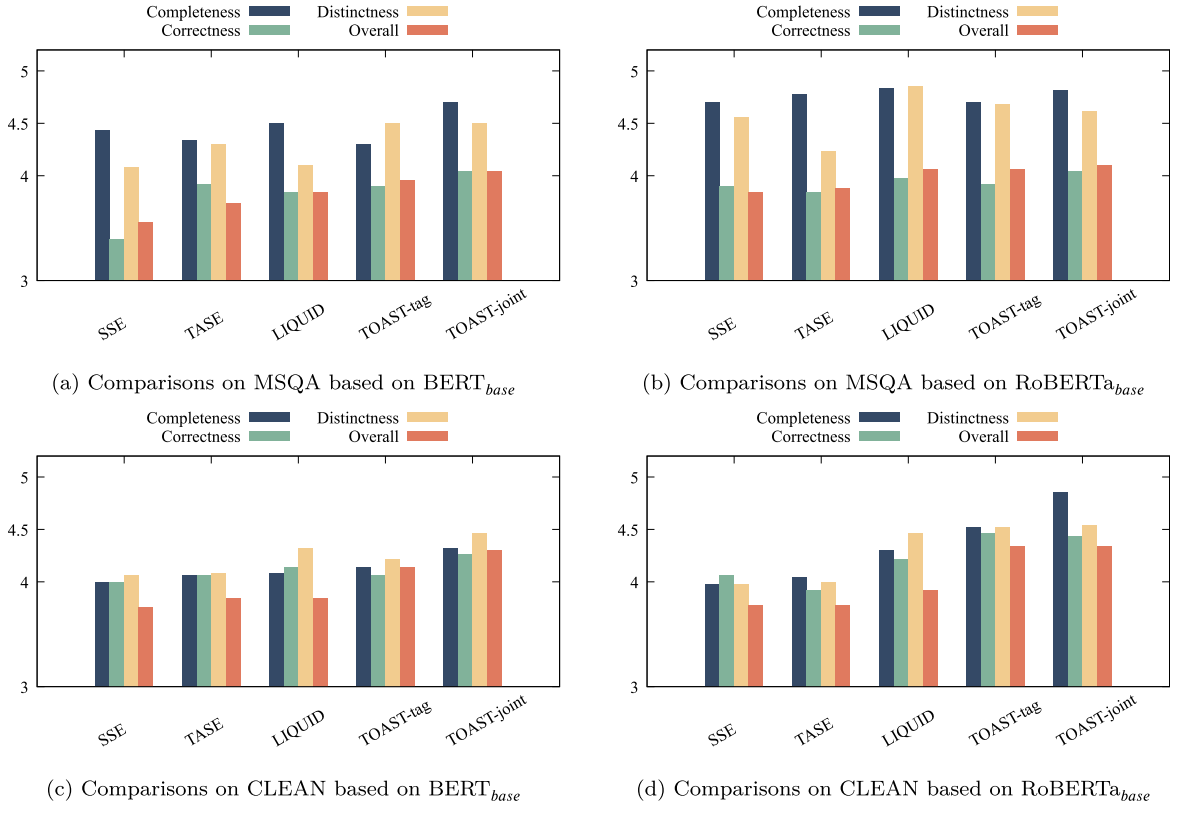
Table 6
The overlapped results for all competing models, including macro-average precision (P), recall (R) and f1 score (F1).

	MSQA			MSQA-Exp			CLEAN		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
[$BERT_{base}$]									
SSE	72.44	48.05	57.77	46.23	42.02	44.03	65.90	67.19	66.54
TASE	77.34	77.25	77.30	70.39	70.03	70.21	67.48	74.02	70.60
SNP-TASE	79.56	73.23	73.23	74.05	68.06	70.47	–	–	–
LIQUID	78.60	79.80	79.20	72.65	71.91	72.29	70.91	67.82	69.33
$TOAST_{tag}$	79.48	78.78	79.12	73.47	71.37	72.40	**82.32**	79.41	80.84
$TOAST_{joint}$	**79.88**	**79.72**	**79.80**	**78.49**	**76.87**	**77.65**	82.16	**79.83**	**80.98**
[$RoBERTa_{base}$]									
SSE	75.65	49.25	59.66	48.53	43.48	45.87	66.24	67.25	66.74
TASE	82.56	81.73	82.14	78.94	**78.81**	78.88	70.81	75.68	73.16
SNP-TASE	80.72	79.83	80.27	47.79	27.10	34.59	–	–	–
LIQUID	80.90	81.50	81.20	78.31	76.14	77.21	75.60	74.96	75.28
$TOAST_{tag}$	83.76	84.27	84.01	79.64	77.72	78.67	83.77	80.58	82.15
$TOAST_{joint}$	**84.06**	**84.40**	**84.23**	**80.65**	78.47	**79.54**	**83.81**	**80.79**	**82.27**

Backed by $BERT_{base}$, $TOAST_{tag}$ achieves EM F1 scores of 63.19 and, 60.70 and 60.05 on the datasets of MSQA, MSQA-Exp and CLEAN, respectively. Moreover, when equipped with the joint decoding strategy, $TOAST_{joint}$ achieves even higher EM F1 scores of 63.53 and, 63.19 and 60.49 on the same datasets. These results showcase substantial improvements over the previous state-of-the-art model, LIQUID, with EM F1 score enhancements ranging from 3.03 to 16.26 across various datasets. Additionally, when utilizing $RoBERTa_{base}$ instead of $BERT_{base}$, $TOAST_{joint}$ achieves EM F1 scores of 71.15, 68.71 and 63.16 on the respective datasets. These scores represent EM F1 improvements of 7.62, 5.52 and 2.67 compared to the previous setup.

The overlapped F1 results shown in Table 6 demonstrate similar trends to the EM F1 results. In comparison to the state-of-the-art (SOTA) model LIQUID, $TOAST_{joint}$ achieves improvements in overlapped F1 scores. Specifically, when using $BERT_{base}$, the improvements are 0.6, 5.36, 11.65 on the MSQA, MSQA-Exp and CLEAN datasets, respectively. Similarly, when employing $RoBERTa_{base}$, the improvements of 3.03, 2.33, and 6.99 are observed on the same datasets. When augmenting TOAST with joint decoding, we observed further performance improvements on both BERT-based and RoBERTa-based encoders. This outcome highlights the effectiveness of the joint decoding strategy, which explicitly combines predictions from multitask modules. Moreover, our proposed model demonstrates robustness in effectively generalizing across different datasets without requiring hyperparameter re-tuning. In contrast, SNP-TASE faces challenge in achieving such generalization. While SNP-TASE demonstrates satisfactory performance when utilizing the $RoBERTa_{base}$ encoder on MSQA dataset, its effectiveness significantly declines when transitioning to MSQA-Exp dataset with the same hyperparameter configuration. Specifically, SNP-TASE achieves an EM F1 score of approximately 27, which is considerably lower compared to other competing models. These results suggest the potential need for hyperparameter re-tuning when applying the SNP-TASE model to different datasets. To ensure a fair comparison with other competing models that did not undergo this re-tuning process, we report the raw results of the SNP-TASE model without any adjustments.

In addition, Fig. 2 presents the comparison results of human evaluation on the overall quality of the complete answer, as well as three aspects at the span-level: completeness, correctness and distinctness. Specifically, Figs. 2(a) and 2(b) illustrate the comparisons on the MSQA dataset using $BERT_{base}$ and $RoBERTa_{base}$, respectively. Similarly, Figs. 2(c) and 2(d) depict the comparisons on the CLEAN dataset. As shown in Fig. 2, the competing models demonstrate superior performance in terms of *completeness* and *distinctness* when extracting spans from the MSQA dataset compared to the CLEAN dataset. This observation can be attributed to the prevalence



(a) Comparisons on MSQA based on BERT_{base}

(b) Comparisons on MSQA based on RoBERTa_{base}

(c) Comparisons on CLEAN based on BERT_{base}

(d) Comparisons on CLEAN based on RoBERTa_{base}

Fig. 2. Human evaluations on MSQA and CLEAN datasets.

of entity-type questions in the MSQA dataset (see Table 4), which are typically shorter and relatively easier to extract completely using tagging models.

Our proposed models, TOAST_{tag} and TOAST_{joint}, demonstrate an overall superior performance in the human evaluation. Specifically, concerning the MSQA dataset, TOAST outperforms the strong baselines TASE and LIQUID in terms of the span correctness and the overall answer quality (see Figs. 2(a) and 2(b)). However, it shows a slight decline in the completeness and distinctiveness aspects of spans when using RoBERTa_{base} (see Fig. 2(b)). We observe that the LIQUID model tends to extract keywords from reference spans of the description type, leading to the fragmentation of answer spans into multiple shorter spans, such as single-word nouns. While these shorter spans are more likely to achieve enhanced completeness and distinctiveness at the span level, but this does not necessarily imply that LIQUID is more effective. In fact, TOAST_{joint} consistently outperforms them in terms of span correctness and the overall quality. On the other hand, in the comparison of different models on the CLEAN dataset, the TOAST models demonstrate marginal improvements across all aspects compared to the competing models. Specifically, the TOAST_{joint} model using RoBERTa_{base} achieves the highest overall score of 4.34 (ranging from 1 to 5), representing improvements of 0.56 and 0.42 over the TASE and LIQUID models, respectively. Similarly, the TOAST_{joint} model using BERT_{base} achieves an overall score of 4.3 which demonstrates improvements of 0.46 over both the TASE and LIQUID models. At the span level, TOAST_{joint} using RoBERTa_{base} exhibit enhancements of 0.64, 0.52, and 0.54 scores of in terms of completeness, distinctness and correctness, respectively. Additionally, TOAST_{joint} using BERT_{base} demonstrates improvements of 0.26, 0.2, and 0.38 scores in these respective aspects.

4.3.2. Case studies

Next, we perform case studies to examine the impact of our proposed methods on answer extraction. The case studies utilize running examples from the MSQA and CLEAN datasets, which are presented in Tables 7 and 8, respectively. In the context, the gold answer spans are annotated in blue color to facilitate identification.

In Table 7, the first example inquires about the proximal attachment of the flexor carpi ulnaris. TASE and LIQUID primarily focus on the flexion and adduction functions of the flexor carpi ulnaris muscle. In contrast, TOAST models accurately identify that the muscle originates from the humeral and ulnar heads, which are its true proximal attachments. The second example explores the prevalent music styles during the Middle Ages. TASE and LIQUID mistakenly categorize the style of composers (e.g. Baroque) as music styles, whereas TOAST models correctly identify all popular music styles. These cases exemplify the effectiveness of TOAST in comprehending the query intention and semantics of the associated context. The third example, which seeks to determine the duration of the Milky Way's rotation, presents a failure case for TOAST. The given context introduces the concept of a galactic

Table 7
The case studies from the MSQA dataset.

Example	Model	Extracted answer
Question: what is the proximal attachment of the flexor carpi ulnaris in humans **Context**: The flexor carpi ulnaris muscle (or FCU) is a muscle of the human forearm that acts to flex and adduct (medial deviation) the hand. The flexor carpi ulnaris muscle arises from two heads, the humeral and ulnar heads, which are connected by a tendinous arch beneath which the ulnar nerve and artery pass. **Gold answer**: [“humeral”, “ulnar”]	SSE	[“humeral and ulnar, connected by a tendinous arch”]
	TASE	[“medial deviation”, “the hand”]
	LIQUID	[“flex”, “medial deviation”, “the hand”, “tendinous”]
	TOAST_{tag}	[“hand”, “humeral”, “ulnar”]
	TOAST_{joint}	[“humeral”, “ulnar”]
Question: popular styles of music in the middle ages **Context**: Medieval music consists of songs , instrumental pieces ... Medieval music was an era of Western music, including liturgical music (also known as sacred) used for the church, and secular music, non-religious music. Medieval music includes solely vocal music, such as Gregorian chant and choral music (music for a group of singers), solely instrumental music ... practice era , a period of shared music writing practices ... **Gold answer**: [“liturgical music”, “secular music”, “Gregorian chant”, “choral music”]	SSE	[“Gregorian chant and choral music”]
	TASE	[“secular music”, “Gregorian chant”, “choral music”, “Baroque”, “Classical music”, “Romantic music”]
	LIQUID	[“Baroque”, “Classical music”, “Romantic”]
	TOAST_{tag}	[“liturgical music”, “secular music”, “Gregorian chant”, “choral music”]
	TOAST_{joint}	[“liturgical music”, “secular music”, “Gregorian chant”, “choral music”]
Question: how long does it take for the milky way to rotate **Context**: The galactic year, also known as a cosmic year, is the duration of time required for the Sun to orbit once around the center of the Milky Way Galaxy. Estimates of the length of one orbit range from 225 to 250 million terrestrial years. The Solar System is traveling at an average speed of 828,000 km/h (230 km/s) or 514,000 mph (143 mi/s) within its trajectory around the galactic center, a speed at which an object could circumnavigate the Earth ’s equator in 2 min and 54 s; ... **Gold answer**: Unanswerable	SSE	[“2 min and 54 s”]
	TASE	[“galactic year”, “million”, “years”, “km/h”, “514,000 mph”, “2 min and 54 s”]
	LIQUID	[“828,000 km/h”, “514,000 mph (143 mi/s)”, “2 min and 54 s”]
	TOAST_{tag}	[“828,000 km/h”, “514,000 mph (143 mi/s)”, “2 min and 54 s”]
	TOAST_{joint}	[“828,000 km/h”, “514,000 mph (143 mi/s)”, “2 min and 54 s”]

year as the time taken for the Sun to complete one orbit around the center of the Milky Way. However, the provided information is insufficient to answer the question. Instead of refusing to answer, we observe that the TASE model extracts the concept of a “galactic year”, albeit with multiple broken spans. Both LIQUID and TOAST models extract information related to the traveling speed of the Solar System and the time it takes to circumnavigate the Earth’s equator at this speed, which pertain to the Solar System rather than the Milky Way. Thus, TOAST and other competing models may struggle to handle cases with intricate semantic relations.

Likewise, the examples in Table 8 from the CLEAN dataset demonstrate similar trends. TOAST models excel in accurately and comprehensively extracting description answers from the given contexts. For instance, in the second example from Table 8, the TASE model tends to fragment the description answer into multiple incomplete spans, whereas TOAST models capture the boundary information of the long description span, thus precisely extracting the entire answer span. The running examples from both datasets emphasize the effectiveness of the proposed model.

5. Related work

In this section, we briefly summarize previous works that are relevant to extractive reading comprehension (RC) datasets and various modeling approaches employed for extractive RC tasks.

5.1. Extractive RC datasets

5.1.1. Single-span RC datasets

Most existing RC datasets, such as SQuAD (Rajpurkar et al., 2016), SQuAD2.0 (Rajpurkar et al., 2018), SearchQA (Dunn et al., 2017), HotpotQA (Yang et al., 2018), TriviaQA (Joshi, Choi, Weld, & Zettlemoyer, 2017) and QuAC (Choi et al., 2018), contain only single-span questions whose answers are limited to a single text span from the provided context, referred as single-span RC datasets. SQuAD consists of passages from Wikipedia as contexts and associated questions whose answers are spans from the passage. SQuAD 2.0 expands SQuAD by adding some questions that are designed to be unanswerable. HotpotQA dataset extends the answer context from single passage to multiple passages. Since the RC datasets like SQuAD or HotpotQA are built by having humans read a given context, write questions and choose a specific answer span, the annotators may tend to make use of words from the answer text which potentially makes their questions easier to answer. To address this concern, one possible solution is to create datasets where questions are not specifically designed with a context in mind. The TriviaQA dataset consists of questions authored by trivia enthusiasts, while the QuAC dataset prevents the annotators (who also serve as questioners) from seeing the full context. However, all of these RC datasets primarily focus on single-span questions, which does not align with the distribution of real-world user questions.

Table 8
The case studies from the CLEAN dataset.

Example	Model	Extracted Answer
Question: 甘肃省哪些城市经济比较发达 (Which cities in Gansu Province have relatively developed economies) **Context**: 论 GDP 人均，嘉峪关，金昌 论 GDP 总量兰州 ，庆阳 (In terms of per capita GDP, Jiayuguan and Jinchang. In terms of total GDP, Lanzhou and Qingyang.) **Gold Answer**: ["嘉峪关", "金昌", "兰州", "庆阳"]	SSE	["嘉峪关，金昌"]
	TASE	["兰州", "庆阳"]
	LIQUID	["嘉峪关，金昌"]
	$TOAST_{tag}$	["嘉峪关", "金昌", "兰州", "庆阳"]
	$TOAST_{joint}$	["嘉峪关", "金昌", "兰州", "庆阳"]
Question: 细菌和古细菌有什么区别 (What are the differences between bacteria and archaea) **Context**: 在细胞结构和代谢上，古菌在很多方面接近其它原核生物... 古菌还具有一些其它特征。与大多数细菌不同，它们只有一层细胞膜而缺少肽聚糖细胞壁。而且，绝大多数细菌和真核生物的细胞膜中的脂类主要由甘油酯组成，而古菌的膜脂由甘油醚构成 ... (In terms of cellular structure and metabolism, ... Archaea also possess some other characteristics. Unlike most bacteria, they only have a single layer of cell membrane and lack peptidogly can cell walls. Additionally, while the cell membranes of most bacteria and eukaryotes are predominantly composed of glycerol esters, the membrane lipids of archaea are composed of glycerol ethers...) **Gold Answer**: ["古菌", "只有一层细胞膜而缺少肽聚糖细胞壁", "细菌和真核生物的细胞膜中的脂类主要由甘油酯组成", "古菌的膜脂由甘油醚构成"]	SSE	["在细胞结构和代谢上，古菌在很多方面接近其它原核生物..."]
	TASE	["大多数细菌", "同", "它们", "只有一层细胞膜而缺少肽聚糖细胞壁", "绝大多数细菌和真核生物", "细胞膜中", "脂类主要由甘油酯组成，而古菌 ", "膜脂由甘油醚构成"]
	LIQUID	["只有一层细胞膜而缺少肽聚糖细胞壁，绝大多数细菌和真核生物的细胞膜中的脂类主要由甘油酯组成,而古菌的膜脂由甘油醚构成"]
	$TOAST_{tag}$	["只有一层细胞膜而缺少肽聚糖细胞壁", "细菌和真核生物的细胞膜中的脂类主要由甘油酯组成", "古菌的膜脂由甘油醚构成"]
	$TOAST_{joint}$	["只有一层细胞膜而缺少肽聚糖细胞壁", "细菌和真核生物的细胞膜中的脂类主要由甘油酯组成", "古菌的膜脂由甘油醚构成"]
Question: 宿州学院的特色专业是什么 (What are the characteristic majors of Suzhou University) **Context**: 会计或英语吧，西校区人力资源管理，地理科学不错 (Accounting or English, human resource management at West Campus, good in geography science) **Gold Answer**: ["会计", "英语", "人力资源管理", "地理科学"]	SSE	["会计或英语"]
	TASE	["会计或英语", "人","管理" ," 地"," 科学"]
	LIQUID	["会计或英语"]
	$TOAST_{tag}$	["会计","英语", "人"]
	$TOAST_{joint}$	["会计", "人"]

5.1.2. Multi-span RC datasets

The complete answer to a real-world user question could consist of multiple text spans. Furthermore, a user question can even have multiple intents, where the answer to each intent is composed of one or more spans. We refer to the datasets focus on such multi-span questions as multi-span RC datasets. Natural Question (Kwiatkowski et al., 2019) and Quoref (Dasigi et al., 2019) both contain multi-span questions. The Natural Question dataset incorporates real anonymized queries to the Google search engine. Quoref is used to validate models with the ability to resolve co-reference among entities. However, the proportion of multi-span answers in Natural Question and Quoref is relatively low, around 2% and 10% respectively. The DROP (Dua et al., 2019) dataset which consists of complex questions on history and football games, requires discrete reasoning over the content of contexts, including co-reference resolution and arithmetic operations, such as addition, sorting and counting. Although the questions could be multi-span, the answer spans are almost exclusively semantically homogeneous and related to numeric values. MASH-QA (Pang et al., 2019) is a domain-specific dataset that extends the answer space to general text types in the healthcare domain. Recently, Li et al. (2022) propose the MultiSpanQA dataset, which consists of open-domain multi-span questions. The MultiSpanQA dataset is derived from Natural Question, whose questions are real queries issued to the Google search engine. Each question is associated with a context

extracted from a retrieved Wikipedia page. MultiSpanQA also has an expanded variant by introducing single-span and unanswerable questions, namely MultiSpanQA (expand).

The above datasets are all in English. The TyDi QA dataset (Clark et al., 2020) offers question–answer pairs in 11 typologically diverse languages, including Arabic, Bengali, Kiswahili, Russian and Thai. The various languages in the dataset bring up new challenges such as morphological variation and word segmentation. To address the scarcity of Arabic datasets for the RC task, the QRCD dataset (Malhas & Elsayed, 2022) provides valuable resources. Additionally, the CMQA dataset (Ju et al., 2022) introduces a new annotate scheme that labels both fine-grained answers and conditions as well as the hierarchical relations in-between. CMQA is the first public multi-span QA dataset in Chinese, however it formulates a new task of conditional question answering. To ameliorate the situation of the lack of Chinese multi-span RC datasets, we propose a Chinese multi-span question answering dataset (CLEAN), which consists of multi-span questions in open domain and supports to cast RC as answer extraction task.

5.2. Neural models for RC

Research in reading comprehension grows rapidly, and many successful neural-based RC models have been proposed in this area. Typically, neural models (Pang et al., 2019; Wang & Jiang, 2017; Xiong, Zhong, & Socher, 2017) for RC are composed of two components, a context encoder and an answer decoder. The context encoder is used to encode the information of questions, contexts and their interactions in-between. Then, the answer decoder aims to generate the answer texts based on outputs of the context encoder. To make the answer decoder compatible with the answer extraction task, Pointer Network (Vinyals, Fortunato, & Jaitly, 2015) model has been adopted to copy tokens from the given contexts as answers (Kadlec, Schmid, Bajgar, & Kleindienst, 2016; Trischler et al., 2016). Wang and Jiang (2017) proposed a boundary model, which utilized Pointer Network to predict the start and end indices for an answer span. Seo, Kembhavi, Farhadi, and Hajishirzi (2017) proposed an alternative way for the implementation of answer decoder, that built neural position classifiers upon the encoder outputs, predicting the start and end indices of the answer span in the context.

Recently, the RC models upgrade the context encoder using pre-trained language models (PrLMs) (Gu et al., 2021; Kenton & Toutanova, 2019; Lee et al., 2020; Liu et al., 2019; Radford, Narasimhan, Salimans, Sutskever, et al., 2018) , benefiting from the invention of Transformer (Vaswani et al., 2017) blocks. Devlin et al. (2019) proposed a standard extractive model for single-span RC that utilizes BERT to encode inputs, then builds position classifiers to predict where the answer span starts and ends. However, the answer decoder, whether implemented with Pointer Network or position classifiers, predicts start and end position independently, thus cannot distinguish the different answer spans properly. Zhu, Ahuja, Juan, Wei, and Reddy (2020) proposed MultiCo which used a contextualized sentence selection method to capture the relevance among multiple sentence-based answer spans in order to form an answer with multiple sentences. These models are not well adapted to multi-span RC which can be formulated as more flexible task of multi-span extraction where each span can be a word, phrase, sentence or any continuous string of text.

Extracting a variable number of spans from an input text can be commonly cast as a sequence tagging problem. Segal et al. (2020) proposed using a sequence tagging model for multi-span extraction, which predicts whether each token is part of an answer. Yoon, Jackson, Lagerberg, and Kang (2022) employed a similar sequence tagging approach to address extractive question answering (Naseem, Dunn, Khushi, & Kim, 2022) in the biomedical domain. Li et al. (2022) also adopted the tagging model architecture, integrating two sub-tasks: predicting the number of spans to extract and annotating the answer structure within their proposed dataset to capture global information. ADRAV (Hu, Yang, Li, Sun, & Yang, 2023) proposed a dynamic routing and answer voting method to further make full use of the hidden layer knowledge of pre-trained models. More recently, LIQUID (Lee et al., 2023) was introduced to automatically generate list-style QA pairs from unlabeled corpora. LIQUID extracted named entities from the summarized text as candidate answers and incorporated synthetic data in the tagging model. These methods harness the extensive factual knowledge embedded in powerful contextualized encoders (PrLMs) or synthesized data that resolves around named entities. As a result, they have demonstrated promising performance in extracting multi-span answers, particularly for factoid questions where the answers correspond entities. However, these methods do not take into account the information of span boundaries in terms of the questions, and thus have very limited capabilities of precisely drawing a description answer. Compared to these approaches, TOAST incorporates the semantic and syntactic information of span boundaries by explicitly modeling the implicit neighboring transitions in-between the adjacent tokens or words, which benefits the span boundary identification.

6. Implications

Our main aim in this study is to address the problem of multi-span question answering, that has widespread implications for real-life applications such as virtual assistants. Although existing reading comprehension (RC) datasets and models have demonstrated effectiveness in answering factoid questions, they encounter difficulties when it comes to handling the complexities inherent in descriptive questions.

We propose the TOAST framework, which is designed for the extraction of multi-span answers, and complement it with a newly created RC dataset named CLEAN that encompasses a significant number of descriptive questions. In Section 4.3, we demonstrate the effectiveness of the components integrated within TOAST. A key insight in the extraction of multi-span answers lies in the value placed on boundary knowledge, particularly through token-based neighboring transitions. To capture this information, we employ an auxiliary task that incorporates token-based transition knowledge, enabling us to jointly learn the sequence tagging task and the transition classification task. Additionally, we further demonstrate the usefulness of the joint decoding strategy in TOAST, which not only implicitly integrates neighboring transition knowledge but also explicitly combines predictions from the enhanced tagging model.

7. Conclusion

In this paper, we propose a joint learning framework named TOAST, which specializes in token-based neighboring transitions to capture the boundary information of answer spans through adjacent word relations for multi-span question answering. Our approach extracts high-quality multi-span answers and is applicable to both alphabet languages like English and logographic languages like Chinese. Furthermore, we introduce an open-domain Chinese multi-span question answering dataset, named CLEAN, that incorporates crafted long answers as contexts. CLEAN effectively bridges the semantic gap by leveraging insights from public contributors, addressing the limitation of existing datasets. Results show that TOAST is more effective than a number of other strong baselines across three publicly available datasets.

8. Limitations

Our proposed approach, TOAST, explores the inherent shifting structures of contexts, incorporating knowledge of span boundaries through token-based transition awareness. However, in contrast to the previous SOTA model LIQUID, it does not utilize external knowledge for answer span extraction. LIQUID improves its performance, particularly for entity-type answer extraction, by integrating open-domain entity knowledge through the augmentation of entity-centric question–answer pairs. Figs. 2(a) and 2(b) show that LIQUID slightly outperforms TOAST in the completeness and distinctiveness aspects of spans on the predominant entity-type question MSQA dataset, attributed to its training with augmented data enriched with entities.

TOAST uniformly addresses spans with varying lengths, be they long descriptive answer spans or shorter entity answer spans, capturing semantic and syntactic shifts between contexts. Note that, such boundary knowledge proves more particularly beneficial for extracting longer answer spans, where such answers are often retrieved as broken spans in other models. As shown in Tables 5–8, TOAST achieves substantial improvements on the CLEAN dataset, which is primarily composed of prevalent descriptive-type questions.

CRediT authorship contribution statement

Zhiyi Luo: Conceptualization, Methodology, Writing – original draft. **Yingying Zhang:** Data curation, Software, Writing – original draft. **Shuyun Luo:** Investigation, Supervision, Writing – review & editing.

Data availability

Data will be made available on request.

Acknowledgments

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), the National Natural Science Foundation of China (Grant No. U22A2004), the Key Research and Development Program of Zhejiang Province, China (Grant No. 2022C01079 and 2023C01041) and the Shaoxing Science and Technology Plan Project (Grant No. 2022B41011).

References

Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W., Choi, Y., et al. (2018). QuAC: Question answering in context. In *Proceedings of the 2018 conference on empirical methods in natural language processing, Brussels, Belgium, October 31 - November 4, 2018* (pp. 2174–2184).

Clark, J. H., Palomaki, J., Nikolaev, V., Choi, E., Garrette, D., Collins, M., et al. (2020). TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions on Association and Computational Linguistics*, *8*, 454–470.

Dasigi, P., Liu, N. F., Marasovic, A., Smith, N. A., & Gardner, M. (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 2019, Hong kong, China, November 3-7, 2019* (pp. 5924–5931).

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (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, NAACL-HLT 2019, minneapolis, MN, USA, June 2-7, 2019, volume 1 (long and short papers)* (pp. 4171–4186).

Dua, D., Wang, Y., Dasigi, P., Stanovsky, G., Singh, S., & Gardner, M. (2019). DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, NAACL-HLT 2019, minneapolis, MN, USA, June 2-7, 2019, volume 1 (long and short papers)* (pp. 2368–2378).

Dunn, M., Sagun, L., Higgins, M., Güney, V. U., Cirik, V., & Cho, K. (2017). SearchQA: A new Q&A dataset augmented with context from a search engine. CoRR arXiv:1704.05179.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., et al. (2021). Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, *3*(1), 1–23.

Hu, Z., Yang, P., Li, B., Sun, Y., & Yang, B. (2023). Biomedical extractive question answering based on dynamic routing and answer voting. *Information Processing & Management*, *60*(4), Article 103367.

Joshi, M., Choi, E., Weld, D. S., & Zettlemoyer, L. (2017). TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th annual meeting of the association for computational linguistics, ACL 2017, vancouver, Canada, July 30 - August 4, volume 1: long papers* (pp. 1601–1611).

Ju, Y., Wang, W., Zhang, Y., Zheng, S., Liu, K., & Zhao, J. (2022). CMQA: A dataset of conditional question answering with multiple-span answers. In *Proceedings of the 29th international conference on computational linguistics, COLING 2022, gyeongju, Republic of Korea, October 12-17, 2022* (pp. 1697–1707).

Kadlec, R., Schmid, M., Bajgar, O., & Kleindienst, J. (2016). Text understanding with the attention sum reader network. In *Proceedings of the 54th annual meeting of the association for computational linguistics, ACL 2016, August 7-12, 2016, berlin, Germany, volume 1: long papers*.

Kenton, J. D. M.-W. C., & Toutanova, L. K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT* (pp. 4171–4186).

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A. P., Alberti, C., et al. (2019). Natural questions: a benchmark for question answering research. *Transactions on Association and Computational Linguistics*, *7*, 452–466.

Lee, S., Kim, H., & Kang, J. (2023). LIQUID: A framework for list question answering dataset generation. arXiv preprint arXiv:2302.01691.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., et al. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, *36*(4), 1234–1240.

Li, H., Tomko, M., Vasardani, M., & Baldwin, T. (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* (pp. 1250–1260).

Liu, Q., Mao, R., Geng, X., & Cambria, E. (2023). Semantic matching in machine reading comprehension: An empirical study. *Information Processing & Management*, *60*(2), Article 103145.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., et al. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

Malhas, R., & Elsayed, T. (2022). Arabic machine reading comprehension on the Holy Qur'an using CL-AraBERT. *Information Processing & Management*, *59*(6), Article 103068.

Naseem, U., Dunn, A. G., Khushi, M., & Kim, J. (2022). Benchmarking for biomedical natural language processing tasks with a domain specific ALBERT. *BMC Bioinformatics*, *23*(1), 144.

Pang, L., Lan, Y., Guo, J., Xu, J., Su, L., & Cheng, X. (2019). HAS-QA: hierarchical answer spans model for open-domain question answering. In *The thirty-third AAAI conference on artificial intelligence, AAAI 2019, honolulu, hawaii, USA, January 27 - February 1, 2019* (pp. 6875–6882). AAAI Press.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.

Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 2: short papers)* (pp. 784–789).

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 2383–2392).

Segal, E., Efrat, A., Shoham, M., Globerson, A., & Berant, J. (2020). A simple and effective model for answering multi-span questions. In *Proceedings of the 2020 conference on empirical methods in natural language processing* (pp. 3074–3080).

Seo, M. J., Kembhavi, A., Farhadi, A., & Hajishirzi, H. (2017). Bidirectional attention flow for machine comprehension. In *5th international conference on learning representations, ICLR 2017, toulon, France, April 24-26, 2017, conference track proceedings*.

Trischler, A., Ye, Z., Yuan, X., Bachman, P., Sordoni, A., & Suleman, K. (2016). Natural language comprehension with the EpiReader. In *Proceedings of the 2016 conference on empirical methods in natural language processing, EMNLP 2016, austin, texas, USA, November 1-4, 2016* (pp. 128–137).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Vinyals, O., Fortunato, M., & Jaitly, N. (2015). Pointer networks. In *Advances in neural information processing systems 28: annual conference on neural information processing systems 2015, December 7-12, 2015, montreal, quebec, Canada* (pp. 2692–2700).

Wang, S., & Jiang, J. (2017). Machine comprehension using match-LSTM and answer pointer. In *5th international conference on learning representations, ICLR 2017, toulon, France, April 24-26, 2017, conference track proceedings*. OpenReview.net.

Xiong, C., Zhong, V., & Socher, R. (2017). Dynamic coattention networks for question answering. In *5th international conference on learning representations, ICLR 2017, toulon, France, April 24-26, 2017, conference track proceedings*. OpenReview.net.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W., Salakhutdinov, R., et al. (2018). HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 2369–2380).

Yoon, W., Jackson, R., Lagerberg, A., & Kang, J. (2022). Sequence tagging for biomedical extractive question answering. *Bioinformatics*, *38*(15), 3794–3801.

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., et al. (2020). Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, *33*, 17283–17297.

Zhu, M., Ahuja, A., Juan, D., Wei, W., & Reddy, C. K. (2020). Question answering with long multiple-span answers. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the association for computational linguistics: EMNLP 2020, online event, 16-20 November 2020* (pp. 3840–3849).