What If Sentence-hood is Hard to Define: A Case Study in Chinese Reading Comprehension

Jiawei Wang[1,2,3], **Hai Zhao**[1,2,3,*], **Yinggong Zhao**[4], **Libin Shen**[4]
[1] Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2] Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University
[3]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
[4]Leyan Tech, Shanghai, China
wjw_sjt@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn
{ygzhao, libin}@leyantech.com

Abstract

Machine reading comprehension (MRC) is a challenging NLP task for it requires to carefully deal with all linguistic granularities from word, sentence to passage. For extractive MRC, the answer span has been shown mostly determined by key evidence linguistic units, in which it is a sentence in most cases. However, we recently discovered that sentences may not be clearly defined in many languages to different extents, so that this causes so-called location unit ambiguity problem and as a result makes it difficult for the model to determine which sentence exactly contains the answer span when sentence itself has not been clearly defined at all. Taking Chinese language as a case study, we explain and analyze such a linguistic phenomenon and correspondingly propose a reader with Explicit Span-Sentence Predication to alleviate such a problem. Our proposed reader eventually helps achieve new a state-of-the-art on Chinese MRC benchmark and shows great potential in dealing with other languages.

1 Introduction

Machine reading comprehension (MRC) is a task that requires models to answer a question according to a given passage. This is a challenging task for it demands to carefully deal with all linguistic granularities from word, sentence to passage (Zhang et al., 2020b; Zhou et al., 2020). For extractive MRC as the focus of this paper, the answer span has been shown mostly determined by key evidence linguistic units, in which it is a sentence in most cases (Zhang et al., 2020a). However, we recently found that sentences may be not clearly defined in many languages to different extents, so that this causes so-called location unit ambiguity problem to let model more difficultly determine which sentence exactly contains the answer span when sentence itself has not been clearly defined at all. In detail, sentence may include multiple clauses like English, or it consists of a series of sub-sentences like Chinese, where all sub-sentences share the same subject, predicate or object (Li et al., 2020b). When a language has relatively strict grammar means to determine the boundaries of sentence constituents such as clauses or sub-sentences, it will facilitate MRC models to more conveniently focus on a certain range of text for finding answer span. Otherwise, there comes an obvious so-called location unit ambiguity problem to hinder the performance of extractive MRC.

In the following, we take Chinese language as a case study to explain and analyze such a linguistic phenomenon and correspondingly find a solution. For the characteristics of Chinese, "*In terms of sentence structure, English is determined by rule, while Chinese is determined by man*" (Wang, 1984), that is, English focuses more on syntax while Chinese focuses more on semantics. A full long English sentence has to be subject to strict grammatical means so that clauses can be clearly identified, while in Chinese, such a long sentence may be written in a loose way, typically, whose subject may be conveniently omitted for all later sub-sentences, so that the boundaries between sentences and sub-sentences are blurred. As a result, there are more independent short sentences in Chinese which may be equally written as a single grammar-rigorous long one in English (Li and Nenkova, 2015; Zhao et al., 2017; Duan and Zhao, 2020).

As shown in a Chinese MRC example in Figure 1, the completely paraphrased sentence to answer the question is given in a series of short sub-

[*] Corresponding author. This paper was partially supported by Key Projects of National Natural Science Foundation of China (U1836222 and 61733011).

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 2348–2359
November 7–11, 2021. ©2021 Association for Computational Linguistics

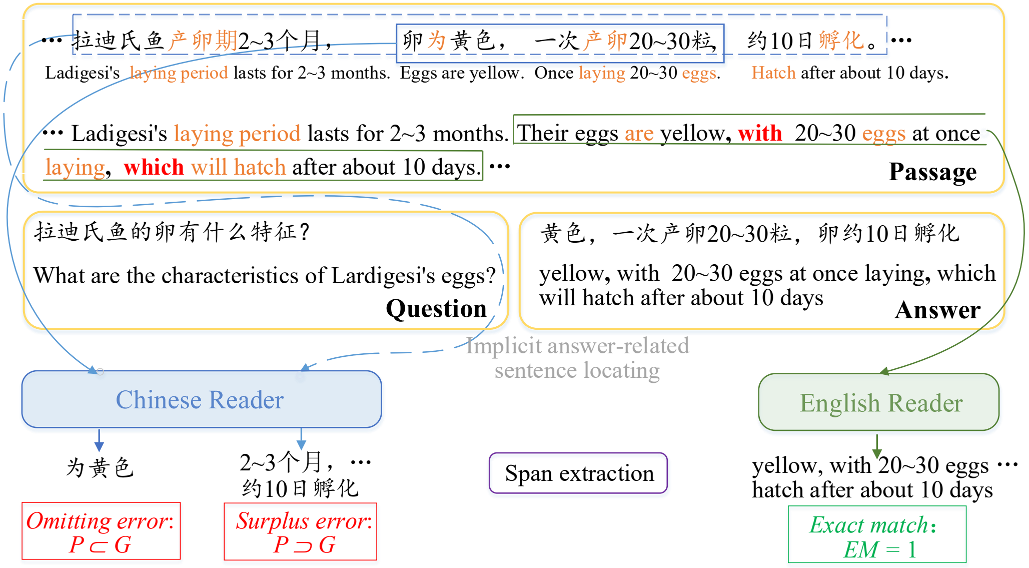

··· 拉迪氏鱼产卵期2~3个月，卵为黄色，一次产卵20~30粒，约10日孵化。···
Ladigesi's laying period lasts for 2~3 months. Eggs are yellow. Once laying 20~30 eggs. Hatch after about 10 days.
··· Ladigesi's laying period lasts for 2~3 months. Their eggs are yellow, with 20~30 eggs at once laying, which will hatch after about 10 days. ···
Passage
拉迪氏鱼的卵有什么特征？
What are the characteristics of Lardigesi's eggs?
Question
黄色，一次产卵20~30粒，卵约10日孵化
yellow, with 20~30 eggs at once laying, which will hatch after about 10 days
Answer
Implicit answer-related sentence locating
Chinese Reader
English Reader
为黄色
2~3个月，···
约10日孵化
Span extraction
yellow, with 20~30 eggs ···
hatch after about 10 days
Omitting error: $P \subset G$
Surplus error: $P \supset G$
Exact match: $EM = 1$

Figure 1: An example of location unit ambiguity of Chinese MRC models compared with English. The main alignment between two languages is marked in orange. P and G refer to predicted span and ground truth answer span, respectively.

sentences in Chinese, which are connected in discourse relation but relatively independent in syntax. Actually, we provide two groups of English translations in Figure 1, in which the same Chinese 'long sentence' may be accurately translated into either a series of short sentences (in small font) or a strictly well-formed long sentence (in big font). In addition to flexible word order, Chinese expressions tend to adopt ellipsis for every possible constituent including the shared subject, leading word or conjunctions, which makes it much more difficult to identify a strictly-defined long sentence in Chinese than in English . Thus assuming that there is a implicit locating process of answer-related sentence before extracting the answer span, English MRC models may easily locate the complete answer-related sentence (right part of Figure 1), while Chinese MRC models may face the location unit ambiguity (left part of Figure 1), ignoring some needed sub-sentences (omitting) or focusing on unrelated ones (surplus). Such specific difficulty in Chinese MRC essentially requires a mechanism that is capable of teaching the model to locate exact answer-related sentences in an explicit way.

In this paper, we intend to discover if this sentence definition difficulty caused location unit ambiguity can be solved well and take a case study on Chinese extractive MRC. The basic form of extractive MRC is requiring models to extract a text span out of the passage to answer the question, given a $\langle passage, question \rangle$ pair, such as SQuAD1.1 (Rajpurkar et al., 2016), NQ (Kwiatkowski et al., 2019) and CMRC 2018 (Cui et al., 2019). There are also some other variants: SQuAD2.0 (Rajpurkar et al., 2018), CoQA (Reddy et al., 2019), HotpotQA (Yang et al., 2018), etc. The mainstream scheme of existing models is modeling extractive MRC as a token-level task, that is, to predict the probability of each token as a start/end span, so as to extract the most suitable answer span (Devlin et al., 2019).

Specifically, we propose ESPReader (Reader with Explicit Span-sentence Predication), applying the proposed extra explicit span-sentence predication (ESP) subtask to help model locate the answer-contained sentences more precisely. ESP is automatically constructed from the original span extraction dataset, which enables the model forcedly to locate the sentence containing the answer span in an expicit way. ESP will be jointly trained with the original token-level task. Our model uses self-attention to acquire answer-aware sentence-level representations from ESP and then fuses them with the original token-level representations from encoder by cross-attention for better span extraction.

Our contribution is summarized as follows:

- To our best knowledge, we are the first to report the sentence definition ambiguity in human language together with its negative impact over MRC task.
- Our proposed ESP can be automatically constructed from the original corpus without extra

human tagging.

- Experiments verify the performance and generality of our proposed model, and a new state-of-the-art on base-level models is achieved.

2 Related Work

2.1 Machine Reading Comprehension

Machine reading comprehension (MRC) is one of the main research directions of natural language processing (NLP). MRC tasks aim at testing machine's comprehension of natural language by requiring to answer questions given a relative passage (Hermann et al., 2015; Zhang et al., 2020d), whose types mainly include cloze (Hill et al., 2015; Cui et al., 2016), multi-choice (Lai et al., 2017; Sun et al., 2019) and span extraction (Rajpurkar et al., 2016; Cui et al., 2019; Reddy et al., 2019). In this paper, we focus on Chinese MRC of the last style. MRC tasks have made great progress and there appeared many models with great performance: Read+Verify (Hu et al., 2019), RankQA (Kratzwald et al., 2019), SG-Net (Zhang et al., 2020c), SAE (Tu et al., 2020), Retro-Reader (Zhang et al., 2021), etc. Among them Reddy et al. (2020) aimed at resolving the partial matched problem in English span extraction tasks, which is close to our model design and task purpose for Chinese. Their solution is constructed as a two-stage model that first locates the initial answer, and then marks it in the raw passage and redoes the reading process. Differently, our method is a fully end-to-end model with a special model design which enables model to learn accurate locations of span-sentences.

2.2 PrLMs and Chinese PrLMs

Pre-trained contextualized language models (PrLMs) like BERT (Devlin et al., 2019) achieved excellent results in various downstream tasks. PrLMs now dominate the encoder design of many NLP tasks, including MRC (Zhang et al., 2021; Xu et al., 2021). More and more well designed PrLMs keep emerging, including XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), ELECTRA (Clark et al., 2020), etc. As for Chinese PrLMs, MacBERT (Cui et al., 2020) uses whole word masking and n-gram masking strategies to select candidate tokens for masking and replaces the [MASK] token with similar words for the masking purpose.

2.3 Multi-grained and Hierarchical Models

To handle the location unit ambiguity, our model quotes a middle-level improvement design, thus our research has some correlation with multi-grained and hierarchical models (Choi et al., 2017; Wang et al., 2018; Luo et al., 2020). Shen et al. (2018) proposed a multi-grained approach combining character-level, word-level and relation-level for text embeddings. Ma et al. (2019) proposed a claim verification framework based on hierarchical attention neural networks to learn sentence-level evidence embeddings to obtain claim-specific representation. All the above works used low-level semantic information to obtain high-level semantic representation, which is different from our intent of using sentence-level information to assist token-level task. Zhang et al. (2020a) proposed a hierarchical network that chooses top K answer-related sentences from the given passage scoring by cosine and bilinear scores to build a new passage for further multi-choice tasks. Their work is somewhat similar to our method. However, we let model directly locate the answer-contained sentence, and use this sentence-level information for further token-level span extraction by cross-attention instead of straightly discarding other lower scoring sentences.

3 Our Proposed Model

As shown in Figure 2, our proposed Reader with Explicit Span-sentence Predication (ESPReader) consists of three modules, that is PrLM encoder, sentence-level self-attention layer and fusion cross-attention layer. The details will be given below.

Explicit Span-sentence Predication To enhance the model with the capacity of locating the answer-related sentences more precisely, an explicit span-sentence predication (ESP) is proposed as a sentence-level subtask. For the sake of the integrity of sentence structure and content, paragraphs are divided into natural sentences by ending punctuation (“,”, “.”, “?”, and “!”) other than a fixed length. After such segmentation, sentence containing the answer span will be labeled as a span-sentence. During training, our model is required to explicitly locate the span-sentence while extracting answer span, which may alleviates the location unit ambiguity issue as span-sentence boundaries have been annotated according to the least sub-sentence segmentation among punctuations.

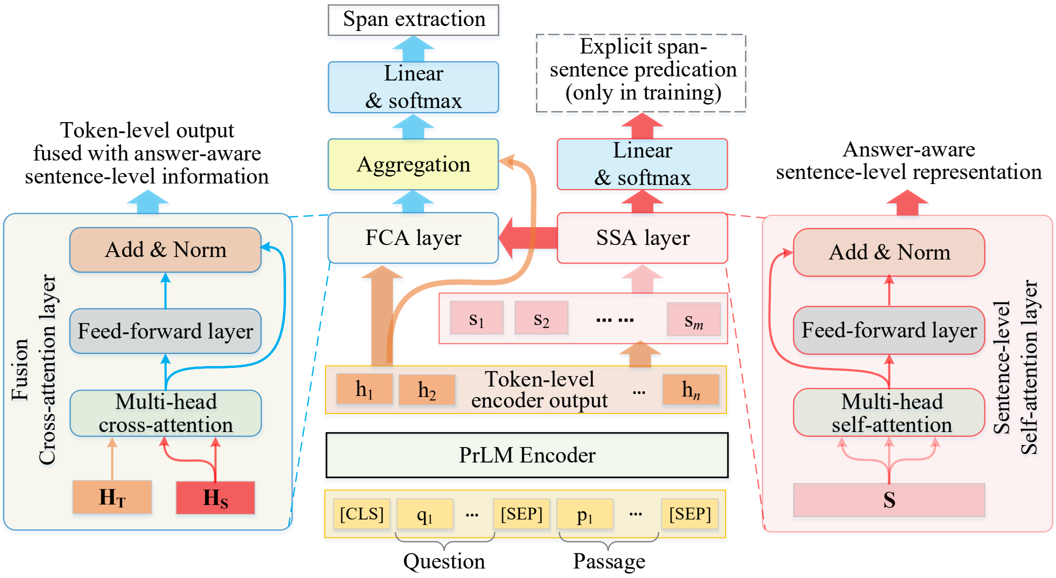

Span extraction
Linear & softmax
Aggregation
FCA layer
Explicit span-sentence predication (only in training)
Linear & softmax
SSA layer
Token-level output fused with answer-aware sentence-level information
Answer-aware sentence-level representation
Fusion Cross-attention layer
Add & Norm
Feed-forward layer
Multi-head cross-attention
H_T
H_S
s_1
s_2
… …
s_m
h_1
h_2
Token-level encoder output
···
h_n
PrLM Encoder
[CLS]
q_1
···
[SEP]
p_1
···
[SEP]
Question
Passage
Sentence-level Self-attention layer
Add & Norm
Feed-forward layer
Multi-head self-attention
S

Figure 2: The architecture of our proposed model.

Since an answer span may stride over multiple sentences, we model the ESP subtask as a form of predicting the location of the start/end sentence (or sub-sentence), which is consistent with the form of original span extraction task.

Sentence Position Embedding We sum up four embeddings including sentence position embedding E_s (see Appendix A for details about E_s) as input to let the PrLM encoder yield representations as: $E_{input} = E_w + E_p + E_t + E_s$, where E_w, E_p and E_t are respectively word embedding, position embedding (the token's offset in the whole input sequence) and token type embedding (the token belongs to question or passage), respectively.

Sentence-level Representation Reimers and Gurevych (2019) found that using mean of the output vector of the last layer of PrLM as sentence representation outperforms the overall representation according to [CLS] token marginally. Li et al. (2020a) claimed that using the average of the last two layers as the sentence embedding is better and mapping it to the standard Gaussian latent space can further eliminate the uneven problem of embedding space caused by word frequency difference.

Taking both experimental effectiveness and model simplicity into consideration, we use the average of last layer's output of PrLM for all tokens $\boldsymbol{H_t} = \{h_t^1, h_t^2, ..., h_t^n\}$ in the corresponding sentences as the sentence-level representation $\boldsymbol{S}$:

$$\boldsymbol{S} = \{s_1, s_2, ..., s_m\}, \\ s_i = \frac{1}{n_i} \sum_{j=sp_i}^{sp_i+n_i-1} h_t^j \tag{1}$$

where sp_i and n_i are the start position offset and length of $sentence_i$, respectively.

Sentence-level Self-attention Layer In terms of the PrLM encoded sentence representations, we apply multi-head attention mechanism (Vaswani et al., 2017) to calculate the self-attention between sentences, as follows:

$$A_s^i = \text{softmax}(\frac{Q_s^i {K_s^i}^T}{\sqrt{d_k}})V_s^i, \\ \tilde{\boldsymbol{H}}_{\boldsymbol{s}} = \textbf{Concate}(A_s^1, A_s^2, ..., A_s^D) \tag{2}$$

where A_s^i is the sentence-level attention score of $head_i$. D is the total number of heads.

$$Q_s^i = \boldsymbol{S}W_s^{Q,i}, \; K_s^i = \boldsymbol{S}W_s^{K,i}, \; V_s^i = \boldsymbol{S}W_s^{V,i} \tag{3}$$

where $W_s^{Q,i}, W_s^{K,i}, W_s^{V,i} \in \mathbb{R}^{d_h \times d_k}$ are all learnable parameters matrices.

Next, $\tilde{\boldsymbol{H}}_{\boldsymbol{s}}$ will be passed through a feed-forward layer followed by GeLU activation (Hendrycks and Gimpel, 2016), and then passed through the residual layer and layer normalization to get the final sentence-level output $\boldsymbol{H_s} = \{h_s^1, h_s^2, ..., h_s^m\}$. To predict the start/end sentence, we use a linear layer with softmax layer to obtain the probability of each sentence as a start/end sentence separately:

$$s_s, s_e = \mathrm{softmax}(\mathrm{Linear}(\boldsymbol{H_s})) \tag{4}$$

where Linear is a linear transformation of $d_h \rightarrow 2$.

Cross entropy loss is used as our training object:

$$L^s = y_{s_s}\mathrm{log}s_s + y_{s_e}\mathrm{log}s_e \tag{5}$$

where y_{s_s} and y_{s_e} are the ground truth label vectors of start/end sentence. Thus, $\boldsymbol{H_s}$ will be guided as answer-aware sentence-level representations.

Fusion Cross-attention Layer To integrate sentence-level information for span extraction, we conduct cross-attention between the output of encoder $\boldsymbol{H_t}$ and the output of sentence-level self-attention layer $\boldsymbol{H_s}$. The calculation is almost the same as Eq. (2), except that the sources of vectors Q, K and V differ: Q comes from $\boldsymbol{H_t}$, while K and V come from $\boldsymbol{H_s}$:

$$\begin{aligned} &Q_F^i = \boldsymbol{H_t} W_F^{Q,i}, \\ &K_F^i = \boldsymbol{H_s} W_F^{K,i},\ V_F^i = \boldsymbol{H_s} W_F^{V,i} \end{aligned} \tag{6}$$

where $W_F^{Q,i}$,$W_F^{K,i}$,$W_F^{V,i}$ are all learnable parameters matrices as Eq. (3). The remaining calculation process is exactly the same as the sentence-level self-attention layer. Through fusion cross-attention layer, the token-level fusion output $\boldsymbol{F_t}$ which is injected with answer-aware sentence-level representations is obtained.

Finally, a manual weight α is used to aggregate $\boldsymbol{F_t}$ and the original encoder output $\boldsymbol{H_t}$ to get the final token-level output:

$$\boldsymbol{H_t'} = \alpha \boldsymbol{H_t} + (1-\alpha)\boldsymbol{F_t} \tag{7}$$

$\boldsymbol{H_t'}$ will be applied to make start/end span predictions t_s and t_e as:

$$t_s, t_e = \mathrm{softmax}(\mathrm{Linear}(\boldsymbol{H_t'})) \tag{8}$$

Equally, cross entropy is used as the token-level loss function:

$$L^t = y_{t_s}\mathrm{log}t_s + y_{t_e}\mathrm{log}t_e \tag{9}$$

where y_{t_s} and y_{t_e} are the ground truth label vectors of start/end span.

Training and Prediction During the training phase, we will jointly learn span extraction and ESP, and the final loss is:

$$\boldsymbol{L} = \beta L^t + (1-\beta)L^s \tag{10}$$

where β is a manual weight.

During the prediction phase, we only make start/end span prediction. The straightforward scoring function is:

$$\textbf{Score}_{raw}(i,j) = t_s^i + t_e^j, \tag{11}$$

where i, j are the start and end token position, respectively ($0 \leq i \leq j \leq n$). Considering that ESPReader is forced to pay more attention to whole sentences by adding the proposed ESP subtask, which might result in a length growth in predicted span, we design a scoring function with inverse length factor (ILF). Note that the span length is not exactly the shorter the better. It only works in this way when two sentences are with the similar length for the sake of reducing redundancy. Taking all these into account, our adopted scoring function is as follows:

$$\begin{aligned} &\textbf{Score}_{ILF}(i,j) = t_s^i + t_e^j + \boldsymbol{ILF}(i,j), \\ &\boldsymbol{ILF}(i,j) = -\mu(j-i)\sqrt{((j-i)/\bar{l}-1)^2} \end{aligned} \tag{12}$$

where $\bar{l}$ is the average length of all candidate answer spans. μ is a manual weight. When the span length is close to the average, ILF will assign some inhibitory effect on long spans. See Appendix B for a more concrete impression on ILF.

	Train	**Dev**	**Test**
Question	10,321	3351	4895
Answer per query	1	3	3
Max passage tokens	962	961	980
Max question tokens	89	56	50
Max answer tokens	100	85	92
Avg passage tokens	452	469	472
Avg question tokens	15	15	15
Avg answer tokens	17	9	9

Table 1: Statistics of the CMRC 2018 dataset.

4 Experiment

4.1 Dataset

Our proposed method is evaluated on the extractive Chinese MRC benchmark, CMRC 2018 (Cui et al.,

	Dev			Test		
Model	EM	F1	Avg	EM	F1	Avg
Human performance*	91.1	97.4	94.3	92.4	97.9	95.2
BERT$_{base}$ *	65.5	84.5	75.0	70.7	87.0	78.9
ELECTRA$_{base}$ *	68.4	84.8	76.6	73.1	87.1	80.1
RoBERTa$_{wwm\_ext\_base}$ *	67.4	87.2	77.3	72.6	89.4	81.0
MacBERT$_{base}$ *	68.5	87.9	78.2	73.2	89.5	81.4
ESPReader						
on BERT$_{base}$	68.7 (↑3.2)	86.3 (↑1.8)	77.5 (↑2.5)	–	–	–
on RoBERTa$_{wwm\_ext\_base}$	70.7 (↑3.3)	88.3 (↑1.1)	79.5 (↑2.2)	–	–	–
on MacBERT$_{base}$	**71.8** (↑3.3)	**88.7** (↑0.8)	**80.3** (↑2.1)	**75.6** (↑2.4)	**90.0** (↑0.5)	**82.8** (↑1.4)
ELECTRA$_{large}$ *	69.1	85.2	77.2	73.9	87.1	80.5
RoBERTa$_{wwm\_ext\_large}$ *	68.5	88.4	78.5	74.2	90.6	82.4
MacBERT$_{large}$ *	70.7	88.9	79.8	74.8	90.7	82.8
MacBERT$_{large\_extData\_v2}$ †	–	–	–	**80.4**	**93.8**	**87.1**
ESPReader						
on RoBERTa$_{wwm\_ext\_large}$	**72.3** (↑3.8)	89.4 (↑1.0)	80.9 (↑2.4)	–	–	–
on MacBERT$_{large}$	**72.3** (↑1.6)	**89.6** (↑0.7)	**81.0** (↑1.2)	77.2 (↑2.4)	91.5 (↑0.8)	84.4 (↑1.6)

Table 2: Results on CMRC 2018. Overall best performances are depicted in boldface (base-level and large-level are marked individually). ↑ refers to the relative increasing compared with according baseline. † refers to unpublished work and the results are gained from CMRC 2018 leaderboard. $*$ refers to results coming from Cui et al. (2020).

2019), which is similar to SQuAD1.1 (Rajpurkar et al., 2016), given a passage, asks model to locate answer span inside for a question, and all questions are supposed to be answerable. The official metrics are Exact Match (EM) and a softer metric F1 score. The dataset details are listed in Table 1 [1].

4.2 Setup

In our ESPReader implementation, we adopt well trained Chinese PrLMs as the encoder. Meanwhile, for each adopted PrLM, we add a one-layer MLP on its top which directly predicts start/end positions of answer span as the default reader to form baseline models for comparison.

We consider three Chinese PrLMs, MacBERT (Cui et al., 2020) which helps achieve the current state-of-the-art on CMRC 2018, Chinese versions of BERT (base[2]) and RoBERTa (base[3] and large[4]).

Our hyperparameters are in Appendix C.

4.3 Results

Table 2 shows the experimental results on CMRC 2018. As can be seen, compared with baselines, our proposed model achieves significant[5] EM and F1 scores improvements on both base-level and large-level models, especially EM scores, with an overall average increase of more than 2%. Moreover, it is noticed that our ESPReader on MacBERT$_{base}$ achieves a new state-of-the-art on CMRC 2018 leaderboard[6] of base-level models by gaining a comparable F1 score to MacBERT$_{large}$, and even outperforming it on EM score on both Dev and Test sets. Besides, ESPReader on MacBERT$_{large}$ also gains the highest EM and average scores among all published work.

4.4 For Different Types Chinese MRC Tasks

To validate the generality of our method, we further test ESPReader on other two different types of Chinese MRC tasks, DRCD (Shao et al., 2018) and CJRC (Duan et al., 2019) (see Appendix D for dataset details). As shown in Table 3, ESPReader obtains visible increase on both datasets compared with our baselines.

[1] There is an extra small Challenge set in CMRC 2018. It especially checks the capability of model reasoning, which is beyond the topic of this work which focuses on the location unit ambiguity. We test our model on this set, which shows an Avg score drop of more than 1%. It is unsurprising and explainable since our ESP task, which forces to locate a certain sentence, might do little help to model's reasoning ability among multiple sentences.

[2] bert-base-chinese

[3] chinese-roberta-wwm-ext

[4] chinese-roberta-wwm-ext-large

[5] we make the McNemar's test (McNemar, 1947) to test the statistical significance of our results. For results in both Tables 2 and 6, we get a p-value<0.01.

[6] `http://ymcui.com/cmrc2018/`

[7] We strictly follow settings provided by Cui et al. (2020) and report the best scores in three times of individual running

	DRCD	CJRC
Model	EM / F1 / Avg	EM / F1 / Avg
	Cui et al. (2020)	
RoBERTa	89.6 / 94.5 / 92.1	62.4 / 82.2 / 72.3
MacBERT	**91.7 / 95.6 / 93.7**	62.9 / **82.5** / 72.7
	Our Implementation [7]	
RoBERTa	88.8 / 94.1 / 91.5	68.6 / 77.5 / 73.1
MacBERT	89.8 / 94.9 / 92.4	70.2 / 79.4 / 74.8
ESPReader		
on RoBERTa	89.6 / 94.7 / 92.2	69.9 / 78.6 / 74.3
on MacBERT	90.3 / 95.1 / 92.7	**71.1** / 80.1 / **75.6**

Table 3: Results on Test set of DRCD and CJRC. RoBERTa and MacBERT refer to chinese-roberta-wwm-ext-large and chinese-macbert-large, respectively.

It is noticed that the improvement on DRCD is not that significant as CJRC (0.3% v.s. 0.8% on MacBERT$_{large}$). One possible explanation is that DRCD is a relatively simple task and the average answer length is 4.9, which means most of the answers are in a single sub-sentence to let our ESP task unnecessary. To validate this, we make statistics on three Chinese MRC datasets to find out the proportions of the examples where a single sub-sentence is sufficient for extracting the answer span, as shown in Table 4. Note that 99.2% examples of DRCD can find answer span in a single sub-sentence, which is consistent with our assumption.

	Needed Sub-sentences		
Dataset	**one**	**two**	**more**
CMRC	74.6%	**13.1%**	**12.3%**
DRCD	**99.2%**	0.7%	0.1%
CJRC	94.7%	3.0%	2.3%

Table 4: Proportions of the examples classified by the number of needed sub-sentences to extract the answer span.

4.5 For Other Languages

Although ESPReader is specifically designed for Chinese MRC, we also test our ESPReader on English MRC benchmarks, SQuAD2.0 (Rajpurkar et al., 2018). The results are shown in Table 5. Even though our model is not supposed to design for English tasks, it still achieves obvious improving compared with two English MRC baseline model (1.8% and 1.6% Avg score, respectively), which indicates our model's potential in dealing with tasks of other languages. To validate this, we further conduct a zero-shot test on Japanese and French datasets provided by Asai et al. (2018), as shown in Tabel 6. On both languages, our method achieves significant improvements, especially the former.

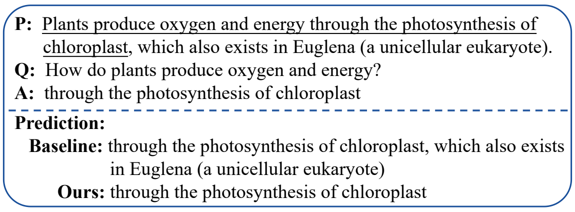
	SQuAD2.0		
Model	EM	F1	Avg
BERT$_{base}$ * [8]	72.6	74.6	73.6
ELECTRA$_{base}$ * [9]	80.9	83.8	82.4
ESPReader *			
on BERT$_{base}$	74.6	76.2	75.4
on ELECTRA$_{base}$	82.5	85.4	84.0

Table 5: Results on Dev set of SQuAD2.0. $*$ refers to our implementation.

	Japanese	French
Model	EM / F1 / Avg	EM / F1 / Avg
Baseline	9.9 / 29.8 / 19.9	25.9 / **45.5** / 35.7
ESPReader	**19.6 / 36.2 / 27.9**	**29.7** / 45.2 / **37.5**

Table 6: Zero-shot test [10] on Japanese and French SQuAD datasets.

Above results indicate that the location unit ambiguity is a common issue in many languages with different seriousness. As shown of an English MRC example in Figure 3, though strictly-defined as a clause of the answer span, sometimes it could be totally unrelated with respect to the question.



P: Plants produce oxygen and energy through the photosynthesis of chloroplast, which also exists in Euglena (a unicellular eukaryote).
Q: How do plants produce oxygen and energy?
A: through the photosynthesis of chloroplast

Prediction:
Baseline: through the photosynthesis of chloroplast, which also exists in Euglena (a unicellular eukaryote)
Ours: through the photosynthesis of chloroplast

Figure 3: An English MRC example with location unit ambiguity. The span-sentence is underlined.

5 Ablation Study

5.1 Effect of Each Module

For the purpose of tracking improvement sources of our ESPReader, we conduct thorough ablation

for each baseline.

[8] bert-base-uncased

[9] electra-base-discriminator

[10] Since Asai et al. (2018) only provided test set for both languages, we fine-tune models on CMRC 2018 (from MacBERT$_{base}$) and SQuAD1.1 (from ELECTRA$_{base}$) and then directly evaluate on Japanese and French, respectively.

studies by adding the proposed modules one by one from the baseline setting (MacBERT$_{base}$). The results are in Table 7. It is noticed that by only adding ESP task, the Avg score on CMRC 2018 Dev set is improved by 1.3%. By further adding sentence-level self-attention layer or fusion cross-attention layer separately, the Avg score does not significantly increase (0.1% and 0.3%, respectively). However, when both of them are included, another visible improvement (0.9%) is obtained.

Model	EM	F1	Avg
Baseline (MacBERT$_{base}$)	68.5	87.9	78.2
+ ESP	70.5	88.3	79.4
+ ESP + SSL	70.9	88.1	79.5
+ ESP + FCL	71.0	88.4	79.7
+ ESP + SSL + FCL	**71.8**	**88.7**	**80.3**
+ ESP + SSL + FCL - SPE	71.5	**88.7**	80.1

Table 7: Results on CMRC 2018 Dev set when adding each module. SSL: sentence-level self-attention layer, FCL: fusion cross-attention layer, SPE: sentence position embedding.

Considering that we additionally introduce sentence position embedding (E_s) on the basis of BERT's embedding layer, we compared the performance of ESPReader with/without E_s. As shown in Table 7, adding E_s can bring a marginal improvement (0.2% Avg score).

Model	EM	F1	Avg
Baseline (BERT$_{base}$)	65.5	84.5	75.0
ESPReader			
+ RS (BERT$_{base}$)	65.8	85.6	75.7
+ ILF (BERT$_{base}$)	**68.7**	**86.3**	**77.5**
Baseline (MacBERT$_{base}$)	68.5	87.9	78.2
ESPReader			
+ RS (MacBERT$_{base}$)	68.7	88.1	78.4
+ ILF (MacBERT$_{base}$)	**71.8**	**88.7**	**80.3**

Table 8: Ablation study results of scoring functions on CMRC 2018 Dev set. RS and ILF refer to **Score**$_{raw}$ and **Score**$_{ILF}$.

5.2 Scoring Function

We keep other settings unchanged and adopt two scoring functions **Score**$_{raw}$ and **Score**$_{ILF}$, respectively. The results are listed in Table 8.

It is observed that the proposed scoring function **Score**$_{ILF}$ makes a nontrivial contribution to the performance of our model on both EM and F1 scores, especially the former (2.9% on BERT$_{base}$ and 3.1% on MacBERT$_{base}$). This observation is in line with our assumption that ESPReader is forced to pay more attention to whole answer-related sentences with an explicit span-sentence predication and thus results in a length growth in predicted span. Note that our model brings increase on both EM and F1 scores to varying degrees, even though ILF is not applied. This indicates that the model benefits from the location guidance produced by ESP task more than suffering from the length growth of predicted span caused by it, which can be well lessened by ILF.

	Error type		
Model	$P \subset G$	$P \supset G$	other
Baseline (MacBERT$_{base}$)	21.7%	57.2%	21.1%
ESPReader (RS)	**19.9%**	63.4%	**16.7%**
ESPReader (ILF)	26.5%	**52.9%**	19.5%

Table 9: General percentage distribution of each error type on CMRC 2018 Dev set.

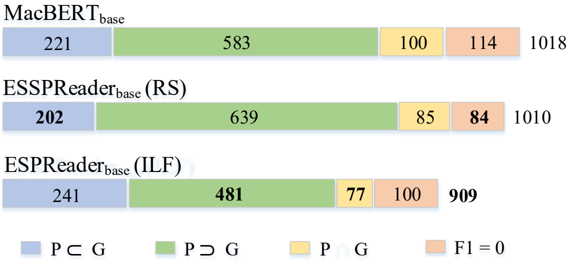
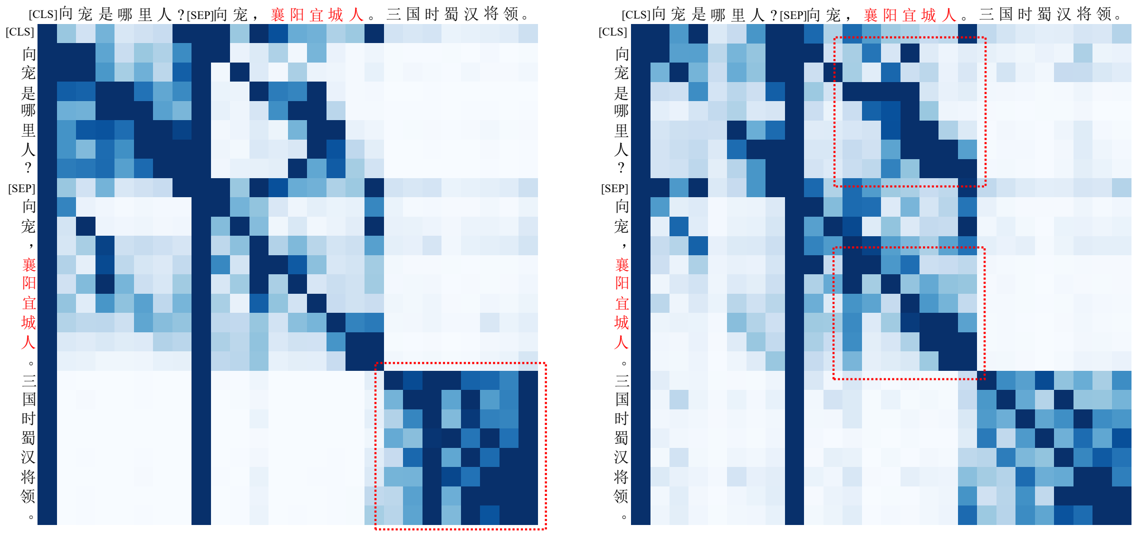


Figure 4: Numbers of each error type on CMRC 2018 Dev set.

5.3 Error Analysis

To take a deep sight into the sources of precision growth, We further research the distribution of each error ($EM = 0$) type after applying our method, the general percentage distribution is shown in Table 9 and the details of actual numbers of each error type are shown in Figure 4. Combining them, we find that with ESP our model decreases both the actual numbers and percentage of all error types (except for the surplus error), of which the actual number of $F1 = 0$ (which means the predicted span is totally unrelated) is dropped by 26.3% (114 → 84). It indicates that ESP effectively corrects the location unit ambiguity issue of Chinese MRC. Note that ILF for scoring helps reduce surplus errors but causes more omitting errors.

In order to have an concrete insight that how



MacBERT

ESPReader

Figure 5: Visualization of the attention scores (average of all heads) of last layers of MacBERT$_{base}$ (left) and ESPReader on MacBERT$_{base}$ (right). The answer span is depicted in red.

our method helps solve location unit ambiguity, we draw attention heatmap of last encoder layer of MacBERT and our proposed model, as shown in Figure 5. Note that the baseline model focuses much on the answer-unrelated sub-sentence. However, the attention distribution of our model is obviously more focused on the span-sentence, which is contributed by our ESP mechanism.

6 Conclusion

This paper aims at addressing the newly discovered difficulty of the boundary ambiguity between sentences and sub-sentences, which exists in many languages to different extents and essentially limits the performance of span extraction MRC models, especially in Chinese environment. We apply explicit span-sentence predication (ESP) to enhance model's ability of precisely locating sentences containing the target span. Our proposed model design is evaluated on Chinese span extraction MRC benchmark, CMRC 2018. The experimental results show that our model significantly improves both EM and F1 scores compared with strong baselines and helps achieve a new state-of-the-art performance. Our method also shows generality and potential in dealing with other languages. This work highlights the research line of further improving challenging MRC by analyzing specific linguistics phenomena.

References

Akari Asai, Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2018. Multilingual extractive reading comprehension by runtime machine translation. *arXiv preprint arXiv:1809.03275*.

Eunsol Choi, Daniel Hewlett, Jakob Uszkoreit, Illia Polosukhin, Alexandre Lacoste, and Jonathan Berant. 2017. Coarse-to-fine question answering for long documents. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 209–220.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for chinese natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP): Findings*, pages 657–668.

Yiming Cui, Ting Liu, Wanxiang Che, Li Xiao, Zhipeng Chen, Wentao Ma, Shijin Wang, and Guoping Hu. 2019. A span-extraction dataset for chinese machine reading comprehension. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5886–5891.

Yiming Cui, Ting Liu, Zhipeng Chen, Shijin Wang, and Guoping Hu. 2016. Consensus attention-based neural networks for chinese reading comprehension. In *Proceedings of the 26th International Conference on*

Computational Linguistics (COLING), pages 1777–1786.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186.

Sufeng Duan and Hai Zhao. 2020. Attention is all you need for chinese word segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3862–3872.

Xingyi Duan, Baoxin Wang, Ziyue Wang, Wentao Ma, Yiming Cui, Dayong Wu, Shijin Wang, Ting Liu, Tianxiang Huo, Zhen Hu, et al. 2019. CJRC: A reliable human-annotated benchmark dataset for chinese judicial reading comprehension. In *China National Conference on Chinese Computational Linguistics*, pages 439–451. Springer.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

KM Hermann, T Kočiskỳ, E Grefenstette, L Espeholt, W Kay, M Suleyman, and P Blunsom. 2015. Teaching machines to read and comprehend. *Advances in Neural Information Processing Systems (NIPS)*, 28.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.

Minghao Hu, Furu Wei, Yuxing Peng, Zhen Huang, Nan Yang, and Dongsheng Li. 2019. Read+Verify: Machine reading comprehension with unanswerable questions. In *Proceedings of the 33th AAAI Conference on Artificial Intelligence (AAAI-19)*, pages 6529–6537.

Bernhard Kratzwald, Anna Eigenmann, and Stefan Feuerriegel. 2019. RankQA: Neural question answering with answer re-ranking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6076–6085.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural Questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics (TACL)*, 7:453–466.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding Comprehension dataset from Examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMCLP)*, pages 785–794.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A Lite BERT for self-supervised learning of language representations. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020a. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130.

Junyi Jessy Li and Ani Nenkova. 2015. Detecting content-heavy sentences: A cross-language case study. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1271–1281.

Zuchao Li, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. 2020b. Explicit sentence compression for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8311–8318.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ying Luo, Fengshun Xiao, and Hai Zhao. 2020. Hierarchical contextualized representation for named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 8441–8448.

Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. 2019. Sentence-level evidence embedding for claim verification with hierarchical attention networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2561–2571.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 784–789.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2383–2392.

Revanth Gangi Reddy, Md Arafat Sultan, Efsun Sarioglu Kayi, Rong Zhang, Vittorio Castelli, and

Avirup Sil. 2020. Answer span correction in machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP): Findings*, pages 2496–2501.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics (TACL)*, 7:249–266.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3973–3983.

Chih Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. 2018. DRCD: a chinese machine reading comprehension dataset. *arXiv preprint arXiv:1806.00920*.

Cun Shen, Tinglei Huang, Xiao Liang, Feng Li, and Kun Fu. 2018. Chinese knowledge base question answering by attention-based multi-granularity model. *Information*, 9(4):2078–2489.

Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019. DREAM: A challenge data set and models for dialogue-based reading comprehension. *Transactions of the Association for Computational Linguistics (TACL)*, 7:217–231.

Ming Tu, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Select, Answer and Explain: Interpretable multi-hop reading comprehension over multiple documents. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI-20)*, pages 9073–9080.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, pages 5998–6008.

Li Wang. 1984. *Chinese grammar theory*. Shandong Education Press, China.

Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1705–1714.

Yi Xu, Hai Zhao, and Zhuosheng Zhang. 2021. Topic-aware multi-turn dialogue modeling. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 5753–5763.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2369–2380.

Shuailiang Zhang, Hai Zhao, Yuwei Wu, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2020a. DCMN+: Dual Co-matching Network for multi-choice reading comprehension. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI-20)*, pages 9563–9570.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020b. Semantics-aware bert for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 9628–9635.

Zhuosheng Zhang, Yuwei Wu, Junru Zhou, Sufeng Duan, Hai Zhao, and Rui Wang. 2020c. SG-Net: Syntax guided transformer for language representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2021. Retrospective reader for machine reading comprehension. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI-21)*.

Zhuosheng Zhang, Hai Zhao, and Rui Wang. 2020d. Machine reading comprehension: The role of contextualized language models and beyond. *Computational Linguistics*, 1.

Hai Zhao, Deng Cai, Yang Xin, Yuzhu Wang, and Zhongye Jia. 2017. A hybrid model for chinese spelling check. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 16(3):1–22.

Junru Zhou, Zhuosheng Zhang, Hai Zhao, and Shuailiang Zhang. 2020. Limit-bert: Linguistics informed multi-task bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP): Findings*, pages 4450–4461.

A Details about Sentence Position Embedding

Sentence position embedding (E_s) indicates the sentence offset of each token. For normal tokens, their sentence positions are the offsets of the segmented sentences they belong to. For special tokens, the sentence position of:

- [CLS]: Set as 0.
- [SEP]: Equal to that of the nearest normal token it follows.
- [PAD]: Set as that of the last normal token plus 1.

In this way, every token is assigned with a sentence position and then a lookup table is used to map these positions to vectors, which is the sentence position embedding.

B ILF Curves

To concretely show the reflections of ILF to different predicted span lengths ($j - i$), we draw curves of ILF value, as shown in Figure 6. It can be seen that ILF achieves the minimum value when span length is equal to the average length of all candidate spans. Besides, ILF has a more obvious inhibitory effect on longer spans than shorter spans.

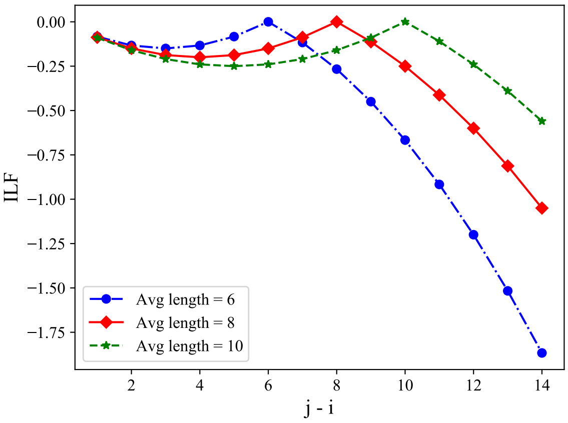


Figure 6: The curves of ILF value under different average lengths of candidate spans when $\mu = 0.1$.

C Settings of Hyperparameters

For the fine-tuning in our tasks in terms of the adopted PrLMs, we set the initial learning rate in {3e-5, 5e-5}. The warm-up rate is set to be 0.1, with a L2 weight decay of 0. The batch size is selected as 24 for base models and 64 for large models. The number of epochs is set to be 2 in all the experiments. Texts are tokenized using wordpieces, with a maximum length of 512 and doc stride of 128. The manual weights are $\alpha = 0.5$, $\beta = 0.8$ and $\mu = 0.1$.

D Details of datasets: DRCD and CJRC

DRCD and CJRC are two different types of Chinese MRC task from CMRC 2018.

- **DRCD**: This is also a span-extraction MRC task but in Traditional Chinese. Besides, compared with CMRC 2018, DRCD contains much more simple questions with short answers and the overall average answer length is 4.9.
- **CJRC**: This is a more complex MRC task, which has yes/no, no-answer and span-extraction questions. This dataset is collected in judicial scenarios. Note that we only use 50% samples of big-train-data.json for training for fair comparison with previous work.