Neural Machine Reading Comprehension: Methods and Trends [*]

Shanshan Liu[†] **Xin Zhang,**[‡] **Sheng Zhang,**[†] **Hui Wang,**[†] **Weiming Zhang**[†]

Science and Technology on Information Systems Engineering Laboratory
College of Systems Engineering
National University of Defense Technology
{liushanshan17,zhangsheng,huiwang,wmzhang}@nudt.edu.cn
Correspondence:ijunzhanggm@gmail.com

Abstract

Machine Reading Comprehension (MRC), which requires the machine to answer questions based on the given context, has gained increasingly wide attention with the incorporation of various deep learning techniques over the past few years. Although the research of MRC based on deep learning is flourishing, there remains a lack of a comprehensive survey to summarize existing approaches and recent trends, which motivates our work presented in this article. Specifically, we give a thorough review of this research field, covering different aspects including (1) typical MRC tasks: their definitions, differences and representative datasets; (2) general architecture of neural MRC: the main modules and prevalent approaches to each of them; and (3) new trends: some emerging focuses in neural MRC as well as the corresponding challenges. Last but not least, in retrospect of what has been achieved so far, the survey also envisages what the future may hold by discussing the open issues left to be addressed.

1 Introduction

Machine Reading Comprehension (MRC) is a task introduced to test the degree to which the machine can understand natural languages via asking the machine to answer questions based on the given context, which can date back to 1970s. Early MRC systems, due to the small size of human-generated datasets and rule-based methods, do not perform well and hence can not be used in practical applications. This situation changes since 2015, which can be attributed to two driving forces. On the one hand, MRC based on deep learning, also called *neural machine reading comprehension*, shows its superiority in capturing contextual information and outperforms traditional rule-based ones dramatically. On the other hand, a variety of large-scale benchmark datasets, such as CNN & Daily Mail [24], SQuAD [64] and MS MARCO [51], make it possible to solve MRC tasks with deep neural architectures and provide testbeds for extensively evaluating the performance of MRC systems. To illustrate the development trends of neural MRC more clearly, we conduct a statistics analysis of representative articles in this field and the result is presented in Fig. 1. As shown in this figure, on the whole, the number of articles increases exponentially from 2015 up till to the end of 2018. Besides, with time going on, the types of MRC tasks come to be increasingly diverse. All of these demonstrate that neural MRC is under rapid development and has become the research focus of both academia (Stanford, Carnegie Mellon, etc.) and industry (Google, Facebook, Microsoft, etc.) evidenced by a large number of authors coming from these institutions.

The flourishing research of neural MRC calls for surveys to systemically study and analyze the recent successes. However, such a comprehensive review still can not be found. Though Qiu et al. [60]
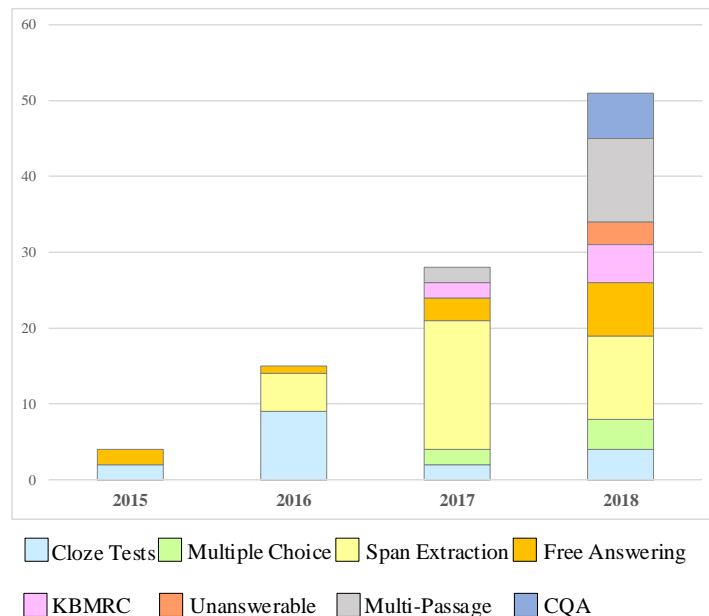
---

[*]Work in progress.

Figure 1: Number of research articles concerned with neural MRC covered in this survey.

give a brief overview very recently to illustrate how to utilize deep learning methods to deal with MRC tasks by introducing several classic neural MRC models, they neither give specific definition of different MRC tasks nor compare each other in depth. Moreover, they do not discuss the new trends and open issues in this field. Motivated by the lack of published surveying effort, we conduct a thorough literature review on recent progresses in neural MRC with the expectation to help researchers, in particular newcomers, to obtain a panoramic view of this field. To achieve that goal, we collect papers mainly using Google Scholar [1] with keywords including *machine reading comprehension*, *machine comprehension*, *reading comprehension*, *deep learning*, and *neural networks*. We select from the searching results only the papers published in related high-profile conferences such as ACL, EMNLP, NAACL, ICLR, AAAI, IJCAI and CoNLL, and restrict the time range to be 2015-2018. In addition, arXiv [2], which includes some latest pre-print articles, is used as a supplementary source. Based on the papers collected, we firstly group common MRC tasks into four types, viz. *cloze tests*, *multiple choice*, *span extraction* and *free answering*, which is in accordance with the categorization of Chen presented in her PhD thesis [5]. We further extend this taxonomy by giving formal definition to each of these types, describing their representative datasets as well as evaluation metrics, and comparing these tasks in different dimensions (Section 2). Secondly, we present the general architecture of neural MRC systems, which consists of four modules including *Embeddings*, *Feature Extraction*, *Context-Question Interaction* and *Answer Prediction* (Section 3). Moreover, the prevalent techniques utilized in each module are also detailed (Section 4). Thirdly, some new trends, such as *knowledge-based MRC*, *MRC with unanswerable questions*, *multi-passage MRC* and *conversational question answering*, are revealed by not only figuring out their challenges but also describing existing approaches and limitations (Section 5). Last but not least, several open issues are discussed with the hope to shed light on possible future research directions (Section 6).

---

[1] http://scholar.google.com

[2] http://arxiv.org/

2 Tasks & Evaluation Metrics

In this section, we introduce various MRC tasks at first, followed by the illustration of evaluation metrics to measure performances of MRC systems according to different tasks.

2.1 Tasks

Following Chen [5], we category MRC problems into four tasks, Cloze Test, Multiple Choice, Span Extraction and Free Answering, mainly based on the answer forms. For better understanding, some examples of representative datasets are presented in Table 1. In the following part, we will give a description of each task with detailed definition.

2.1.1 Cloze Test

Cloze tests, also known as gap-filling tests, are commonly adopted in exams to evaluate students' language proficiency. Inspired by that, this task is utilized to measure the ability of machines in natural language understanding. In cloze tests, questions are generated by removing some words or entities from the passage. To answer questions, it is asked to fill in the blank with the missing ones. Some tasks provide candidate answers, but it is optional. Cloze tests, which add obstacles to reading, require understanding of context and usage of vocabulary and are challenging for machine reading comprehension. The most prominent feature of cloze tests is that answers are words or entities in the context and this task can be regarded as words or entities prediction.

> **Cloze Tests**
>
> Given the context C, from which a word or an entity $a(a \in C)$ is removed, the cloze tests ask the model to fill in the blank with the right word or entity a by maximizing the conditional probability $P(a|C - \{a\})$.

- CNN & Daily Mail

This dataset, built by Hermann et al. [24], is one of the most representative cloze-style MRC datasets. CNN & Daily Mail, consisting of 93,000 articles from CNN and 220,000 articles from Daily Mail, is indeed large-scale and makes it possible to utilize deep learning approaches in MRC. Considering that bullet points are abstractive and have little sentence overlap with documents, Hermann et al. replace one entity at a time with a placeholder in these bullet points and evaluate the machine reading system by asking machine to read the documents and then predict which entity the placeholder in bullet points refers to. As questions are not proposed directly from documents, this task is challenging and some information extraction methods fail to deal with it. This methodology of creating MRC datasets enlightens lots of other researches[77, 52, 69]. In order to avoid that questions can be answered by knowledge out of the documents, all entities in documents are anonymized by random markers.

- CBT

Hill et al.[25] design the cloze-style MRC dataset, CBT (The Children's Book Test), from another perspective. They collect 108 children's books and form each sample with 21 consecutive sentences from chapters in those books. To generate questions, a word from 21st sentence is removed and the other 20 sentences act as context. Nine incorrect words, whose type is as same as the answer, are selected at random from the context as candidate answers. There are some differences between the CNN & Daily Mail and the CBT. Firstly, unlike the CNN & Daily Mail, entities in the CBT are not anonymized so that models can utilize background knowledge from wider context. Secondly, missing items in the CNN & Daily Mail are limited to named entities, but in the CBT there are four distinct types: named entities, nouns, verbs and prepositions. Thirdly, the CBT provides candidate answers, which simplifies the task in a way. Overall, with appearance of the CBT, context, which plays a significant role in human comprehension, has gained much more attention.

Considering that more data can significantly improve performance of neural network models, Bajgar et al. [3] introduce the BookTest, which enlarges the CBT dataset 60 times and enables training larger models.

- LAMBADA

Table 1: A few examples of MRC datasets

Cloze Test		
CLOTH[93]	Context:	Comparisons were drawn between the development of television in the 20th century and the diffusion of printing in the 15th and 16th centuries. Yet much had happened __1__. As was discussed before, it was not __2__ the 19th century that the newspaper bec -ame the dominant pre-electronic __3__, following in the wake of the pamphlet and the book and in the __4__ of the periodical.· · ·
	Options:	1. A.between B.before C.since D.later 2. A.after B.by C.during D.until 3. A.means B.method C.medium D.measure 4. A.process B.company C.light D.form
	Answer:	1.A 2.D 3.C 4.B
Multiple Choice		
RACE[36]	Context:	If you have a cold or flu, you must always deal with used tissues carefully. Don't leave dirty tissues on your desk or on the floor. Someone else has to pick these up and viruses could be passed on.
	Question:	Dealing with used tissues properly is important because _____.
	Options:	A. it helps keep your classroom tidy B. people hate picking up dirty tissues C. it prevents the speard of colds and flu D. picking up lots of tissues is hard work
	Answer:	C
Span Extraction		
SQuAD[64]	Context:	Computational complexity theory is a branch of the theory of computation in theoretical computer science that focuses on classifying computational problems according to their inherent difficulty, and relating those classes to each other. A computa -tional problem is understood to be a task that is in principle amenable to being solved by a computer, which is equivalent to stating that the problem may be solved by mechanical application of mathematical steps, such as an algorithm.
	Question:	By what main attribute are computational problems classified utilizing computational complexity theory?
	Answer:	inherent difficulty
Free Answering		
MS MARCO[51]	Context 1:	Rachel Carson's essay on The Obligation to Endure, is a very convincing argument about the harmful uses of chemical, pest -icides, herbicides and fertilizers on the environment.
	. . . . . .	
	Context 5:	Carson believes that as man tries to eliminate unwanted insects and weeds, however he is actually causing more problems by polluting the environment with, for example, DDT and harming · living things
	. . . . . .	
	Context 10:	Carson subtly defers her writing in just the right writing for it to not be subject to an induction run rampant style which grabs the readers interest without biasing the whole article.
	Question:	Why did Rachel Carson write an obligation to endure?
	Answer:	Rachel Carson writes The Obligation to Endure because believes that as man tries to eliminate unwanted insects and weeds, howe -ver he is actually causing more problems by polluting the enviro -nment.

To take the meaning of the wider context into account, Paperno et al. [56] propose the LAM-BADA dataset (**LA**nguage **M**odeling **B**oardened to **A**ccount for **D**iscourse **A**spects). Similar to the CBT, the source of LAMBADA is also books and the task is word prediction. However, the target word which needs to be predicted in the LAMBADA is the last word in the target sentence while in the CBT any word in the target sentence may be targeted. Moreover, Paperno et al. find that some samples in the CBT can be guessed just with the target sentence alone rather than wider context. To overcome this shortcoming, there is a constraint in the LAMBADA that it is difficult to predict the target word correctly only with the target sentence. That is to say, comprared with the CBT, the LAMBADA requires more understanding of wider context.

- Who-did-What

In order to better evaluate the understanding of natural language, researchers try to avoid sentence overlap between questions and documents when constructing MRC datasets. Onishi et al. [52] provide a new insight into how to reduce the syntactic similarity. In the "Who-did-What" dataset, each sample is formed from two independent articles. One servers as the context and questions are generated from the other. This approach can be utilized by other corpus, in which articles don't have summary points unlike CNN & Daily Mail. There is another feature of the Who-did-What, just as shown in the name, the dataset only pays attention to the person name entity, which may be its limitation.

- CLOTH

Different from above automatically-generated datasets, CLOTH [93] (**CLO**ze test by **T**eac**H**ers) is human-created, which is collected from English exams for Chinese students. Questions in the CLOTH are well-designed by middle-school and high-school teachers to examine students' language proficiency including vocabulary, reasoning and grammar. There are less purposeless or trivial questions in the CLOTH so that it requires a deep understanding of language.

- CliCR

To address the problem that there are scarce datasets for specific domains, Suster et al.[77] build a large-scale cloze-style dataset based on clinical case reports for healthcare and medicine. Similar to the CNN & Daily Mail, summary points of each case reports are used to create queries by blanking out a medical entity. The introduction of CliCR promotes the application of MRC in practical use like clinical decision.

2.1.2 Multiple Choice

Multiple choice is another machine reading comprehension task inspired by language proficiency exams. It is required to select the right answer to the question from candidates according to the provided context. Compared to cloze tests, answers for multiple choice are not limited to words or entities in the context, so the answer form is more flexible. But, it is a must for this task to provide candidate answers.

> **Multiple Choice**
>
> Given the context C, the question Q and a list of candidate answers $A = \{a_1, a_2, \cdots, a_n\}$, the multiple choice task is to select the right answer a_i from A $(a_i \in A)$ by maximizing the conditional probability $P(a_i|C, Q, A)$.

- MCTest

MCTest, proposed by Richardson et al. [66], is a multiple choice machine reading comprehension dataset at the early stage. It consists of 500 fictional stories, and for each story there are four questions with four candidate answers. Choosing fictional stories is to avoid introducing external knowledge, and questions can be answered according to the given story itself. This idea of using story-based corpus inspires other datasets, such as CBT [25] and LAMBADA [56]. Although the appearance of MCTest encourages research of machine reading comprehension, its size is too small so that it is not suitable for some data-hungry techniques.

- RACE

Like CLOTH dataset [93], RACE [36] is also collected from the English exams for middle school and high school Chinese students. This corpus allows types of passages to be more various. In contrast to one fixed style for the whole dataset, such as news for CNN & Daily Mail

[24] and NewsQA [80], fictional stories for CBT [25] and MCTest [66], almost all kinds of passages can be found in RACE. As a multiple choice task, RACE asks for more reasoning because questions and answers are human-generated and simple methods based on information retrieval or word co-occurrence may not perform well. In addition, compared to MCTest [66], RACE contains about 28,000 passages and 100,000 questions, which is large-scale and supports the training of deep learning models. All aforementioned features illustrate that RACE is well-designed and full of challenges.

2.1.3 Span Extraction

Although cloze tests and multiple choice can measure the ability of machine in natural language understanding to some extent, there are limitations in those tasks. To be more concrete, words or entities are not sufficient to answer questions. Instead, some complete sentences are required. Moreover, there are no candidate answers in many cases. Span extraction task can well overcome above weaknesses. Given the context and question, this task asks the machine to extract a span of text from the corresponding context as the answer.

Span Extraction

Given the context C, which consists of n tokens, that is $C = \{t_1, t_2, \cdots, t_n\}$, and the question Q, the span extraction task asks to extract the continuous subsequence $a = \{t_i, t_{i+1}, \cdots, t_{i+k}\}(1 \leq i \leq i + k \leq n)$ from context C as the right answer to question Q by maximizing the condition probability $P(a|C, Q)$.

- SQuAD

SQuAD (Stanford Question Answering Dataset), proposed by Rajpurkar et al.[64] of Stanford University, can be regarded as a milestone for machine reading comprehension. With SQuAD dataset being released, a machine reading comprehension competition based on that gradually draws attention of both academia and industry, which in turn stimulates the appearance of various advanced MRC techniques.

Collecting 536 articles from Wikipedia, Rajpurkar et al. require crowd-workers to pose more than 100,000 questions and select a span of arbitrary length from the given article to answer the question. SQuAD is not only large but also in high quality. In contrast to prior datasets, SQuAD defines a new kind of MRC task, which does not provide answer choices and needs a span of text as the answer rather than a word or an entities.

- NewsQA

NewsQA [80] is another span extraction dataset similar to SQuAD, in which questions are also human-generated and answers are spans of text from corresponding articles. The obvious difference between NewsQA and SQuAD is the source of articles. In NewsQA, articles are collected form CNN news while the SQuAD is based on Wikipedia. It is worth mentioning that some questions in NewsQA have no answer according to the given context. The addition of unanswerable questions makes it closer to reality and inspires Rajpurkar et al. [63] to update SQuAD to version 2.0. In terms of unanswerable questions, we will give a detailed introduction in the section 5.2.

- TriviaQA

The construction process of TriviaQA [32] distinguishes itself from previous datasets. In prior work, crowd-workers are given articles at first and pose the questions closely related to those articles. However, this process results in the dependence of questions and evidences to answer them. Furthermore, in human understanding process, people often ask a question in the first place and then find useful resources to answer it. To overcome this shortcoming, Joshi et al. firstly gather question-answer pairs from trivia and quiz-league websites. Then they search for evidence to answer questions from webpages and Wikipedia. Finally, over 650,00 question-answer-evidence triples are built for machine reading comprehension task. This novel construction process makes TriviaQA a challenging testbed with considerable syntactic variability between questions and contexts.

- DuoRC

Saha et al. [69] also try to reduce lexical overlap between questions and contexts in DuoRC. Like Who-did-What [52], questions and answers in DuoRC are created from two different versions

of documents corresponding to the same movie, one from Wikipeidia and another from IMDb. Asking questions and labelling answers are done by different group of crowd workers. The distinction between two versions of movie plots asks for more understanding and reasoning. Moreover, there are unanswerable questions in DuoRC.

2.1.4 Free Answering

Compared to cloze tests and multiple choice, span extraction task makes great strides in promoting machines to give more flexible answers, yet it is not enough, for that answers restricted to a span of the context is still unrealistic. To answer the questions, the machine needs to reason across multiple pieces of the context and summarize the evidence. Among these four tasks, free answering is the most complicated one as there is no limitations to its answer forms and it is more suitable for real application scenarios.

Free Answering

Given the context C and the question Q, the right answer a in free answering task may not be subsequence in the original context C, namely either $a \subseteq C$ or $a \nsubseteq C$. The task asks to predict the right answer a by maximizing the conditional probability $P(a|C, Q)$.

Different from the other tasks, free answering reduces some constraints and pays much more attention to utilizing free-form natural language to better answer questions.

- bAbI

bAbI, proposed by Weston et al. [91], is a well-known synthetic machine reading comprehension dataset. It consists of 20 tasks, generated with a simulation of a classic text adventure game. Each task is independent from others and tests one aspect of text understanding, such as recognizing two or three argument relations, using basic deduction and induction. Weston et al. think that dealing with all these tasks is a prerequisite to full language understanding. Answers are limited to a single word or a list of words and may not be directly found from original context. The release of bAbI dataset promotes the development of several promising algorithms, but as all data of bAbI is synthetic, it is a little far away from the real world.

- MS MARCO

MS MARCO [51] can be viewed as another milestone of machine reading comprehension after SQuAD [64]. To overcome weaknesses of previous datasets, it has four predominant features. Firstly, all of questions are collected from real user queries. Secondly, for each question, ten related documents are searched from the Bing search engine to serve as the context. Thirdly, labelled answers to those questions are generated by human so that they are not restricted to spans of the context and more reasoning and summarization are required. Fourthly, there are multiple answers to one question and sometimes they are even conflicted, which is more challenging for the machine to select the right answer. The proposal of MS MARCO makes machine reading comprehension dataset closer to real world.

- SearchQA

The work of SearchQA [21] is just like TriviaQA [32], both of them follow the general pipeline of question-answering. To construct SearchQA, Dunn et al. firstly collect question-answer pairs from J!Archive and then search for snippets related to questions from Google. However, the major difference between SearchQA and Trivia QA is that in TriviaQA there is one document with evidence for each question-answer pair while in SearchQA each pair has 49.6 related snippets on average.

- NarrativeQA

Seeing the limitation that evidence for answering the question just from a single sentence of original context in most previous datasets, Kovcisky et al. [35] design the NarrativeQA. Based on book stories and movie scripts, they search related summaries from Wikipedia and ask co-workers to generate question-answer pairs according to those summaries. What makes the NarrativeQA special is that answering questions requires understanding of the whole narrative, rather than superficial matching information.

- DuReader

Similar to MS MARCO [51], DuReader, released by He et al. [23] is another large-scale machine reading comprehension dataset from real world application. Questions and documents in DuReader are collected from Baidu Search (search engine) and Baidu Zhidao (question answering community). Answers are human generated instead of spans in original contexts. What makes DuReader different is that it provides new question types such as yes-no and opinion. Compared to factoid questions, the new ones sometimes require summary over multiple parts of documents, which leaves opportunity for research community.

2.1.5 Comparison of different tasks

To compare the contribution and limitation of four MRC tasks, we conduct the comparison in five dimensions: construction, understanding, flexibility, evaluation and application. For each dimension, its score varies from 1 to 4 according to their relative rankings, and the higher score, the better performance in that dimension.

- Construction: This dimension measures whether it is easy to construct datasets for the task or not. The easier, the higher score.
- Understanding: This dimension evaluates how well the task can test the understanding ability of machines. If the tasks need more understanding and reasoning, the score of this dimension is higher.
- Flexibility: The flexibility of answer form can measure the quality of tasks. When answers are more flexible, the flexibility score is higher.
- Evaluation: Evaluation is a necessary part of MRC tasks. Whether a task can be easily evaluated also determines its quality. The task which is easy to be evaluated gets high score in this dimension.
- Application: A good task is supposed to be close to real world application. So score of this dimension is high, if a task can be applied to real world easily.
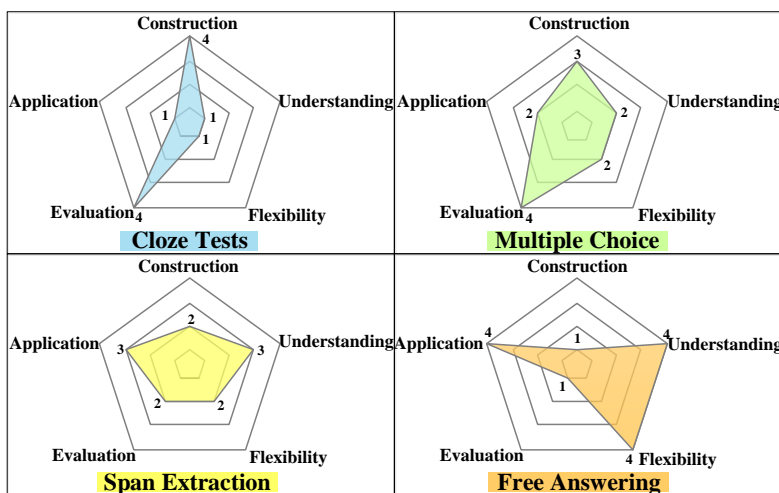

Figure 2: Comparison of different MRC tasks.

As presented in Fig. 2, scores of these five dimensions vary from different tasks. To be more concrete, cloze test tasks are the easiest to construct datasets and be evaluated. However, as the answer forms are restricted to single word or name entity in the original context, cloze tests cannot test the understanding of machines well and are inconformity with the real world application. Multiple choice tasks provide candidate answers for each question so that even if answers are not limited in the original context, they can be easily evaluated. It is not very hard to build datasets for this task as multiple choice tests in language exams can be easily utilized. However, candidate answers lead to a gap between synthetic datasets and realistic application. In contrast, span extraction tasks are

a moderate choice, for which the datasets are easy to be constructed and be evaluated. Moreover, they can test machine's understanding of text in a way. All of these advantages contribute to quite a lot research focusing on these tasks. The disadvantage of span extraction is that answers are constrained to the subsequence of original context, which is still a little far away from real world. Free answering tasks show their superior in understanding, flexibility and application dimensions, which are the closest to practical application. However, every coin has two sides. Because of the flexibility of its answer form, it is hard to build datasets somewhat and how to effectively evaluate performance on these tasks remains an exclusive challenge.

2.2 Evaluation Metrics

For different MRC tasks, there are various evaluation metrics. To evaluate cloze tests and multiple choice tasks, the most common metric is accuracy. In terms of span extraction, exact match (EM), a variant of accuracy, and F1 score are computed to measure performance of models. Considering that answers for free answering tasks are not limited to the original context, ROUGE-L and BLEU are wildly utilized. In the following part, we will give a detailed illustration of these evaluation metrics.

- Accuracy

Accuracy with respect to the ground-truth answers is usually applied to evaluate cloze tests and multiple choice tasks. When given a question set $Q = \{Q_1, Q_2, \cdots Q_m\}$ with m questions, if the model correctly predicts answers for n questions, then accuracy calculates as follows:

$$\text{Accuracy} = \frac{n}{m}. \tag{1}$$

Exact match is a variant of accuracy which evaluates whether a predicted answer span matches the ground truth sequence exactly or not. If the predicted one is equal to the gold one, the value of EM will be 1 and 0 otherwise. It can also be calculated by above equation.

- F1 Score

F1 score is a common metric in classification tasks. In terms of MRC, both candidate answers and reference answers are treated as bag of tokens and true positive (TP), false positive (FP), true negative (TN) and false negative (FN) are denoted as shown in Table 2.

Table 2: The definition of TP, TN, FP, FN.

	tokens in reference	tokens not in reference
tokens in candidate	**TP**	**FP**
tokens not in candidate	**FN**	**TN**

Then the precision and recall are computed as below:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{2}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{3}$$

F1 score, also known as balanced F score, is the harmonic average of precision and recall:

$$\text{F1} = \frac{2 \times P \times R}{P + R}, \tag{4}$$

where P denotes precision while R is recall.

Compared to EM, this metric loosely measures the average overlap between the prediction and the ground truth answer.

- ROUGE-L

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is an evaluation metric initially for automatic summarization, proposed by Lin and Chin-Yew [39]. It evaluates the quality of a summary by counting the number of overlapping between model-generated one and ground-truth.

There are various ROUGE measures, ROUGE-N, ROUGE-L,ROUGE-W,ROUGE-S,etc., adapted to different evaluation requirements, among which ROUGE-L is wildly utilized in MRC tasks with the appearance of free answering. Unlike other metrics, such as EM or accuracy, ROUGE-L is more flexible which mainly measures the similarity between gold answer and predicted one. "L" in ROUGE-L denotes the longest common subsequence (LCS) and ROUGE-L can be computed as follows:

$$R_{lcs} = \frac{LCS(X,Y)}{m}, \tag{5}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n}, \tag{6}$$

$$F_{lcs} = \frac{(1+\beta)^2 R_{lcs} P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}}, \tag{7}$$

where X is ground-truth answer with m tokens, Y is model-generated answer with n tokens, and $LCS(X,Y)$ denotes the length of the longest common subsequence of X and Y.

Using ROUGE-L to evaluate performance of MRC models does not require predicted answers to be consecutive subsequence of the ground-truth, whereas the more token overlap contributes to the higher ROUGE-L score. However, length of candidate answers has an effect on the value of ROUGE-L.

- BLEU

BLEU (Bilingual Evaluation Understudy), proposed by Papineni et al. [57], is widely used to evaluate the translation performance at first. When adapted to MRC tasks, BLEU score measures the similarity between predicted answers and ground truth. The cornerstone of this metric is precision measure, which is calculated as follows:

$$P_n(C,R) = \frac{\sum_i \sum_k \min(h_k(c_i), \max(h_k(r_i)))}{\sum_i \sum_k h_k(c_i)}, \tag{8}$$

where $h_k(c_i)$ counts the number of k-th n-gram appearing in candidate answer c_i, in a similar way, $h_k(r_i)$ denotes the occurrence number of that n-gram in gold answer r_i.

For the value of $P_n(C,R)$ is higher when answer spans are shorter, such precision cannot measure the similarity well solely. The penalty factor BP is introduced to alleviate that, which is computed as:

$$\text{BP} = \begin{cases} 1, l_c > l_r \\ e^{1-\frac{l_r}{l_c}}, l_c \leq l_r. \end{cases} \tag{9}$$

Finally, the BLEU score is computed as below:

$$\text{BLEU} = \text{BP} \cdot \exp(\sum_{n=1}^{N} w_n \log P_n), \tag{10}$$

where N means using n-grams up to length N and w_n equals $1/N$. The BLEU score is the weighted average of each n-gram and the maximum of N is 4, namely BLEU-4.

BLEU score can not only evaluate the similarity between candidate answers and ground-truth answers but also test the readability of candidates.

3 General Architecture

As presented in Fig. 3, a typical machine reading comprehension system, which takes the context and question as inputs and outputs the answer, contains four key modules: Embeddings, Feature Extraction, Context-Question Interaction and Answer Prediction. The function of each module can be interpreted as follows:
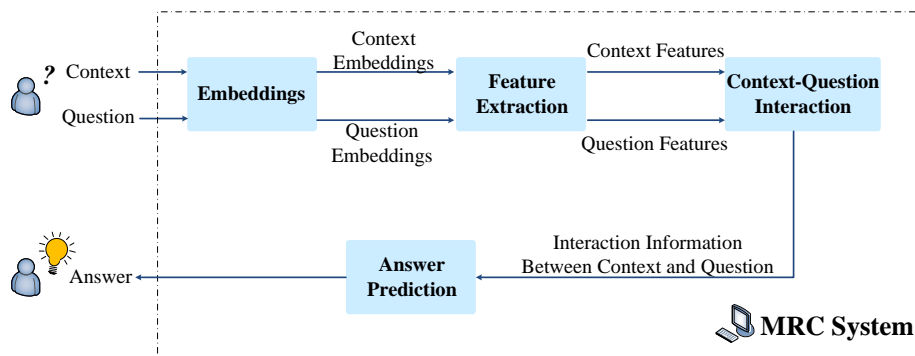
Figure 3: The general architecture of machine reading comprehension system.

- Embeddings: As the machine is unable to understand natural language directly, it is indispensable for the Embedding module to change input words into fixed-length vectors at the beginning of the MRC systems. Taking the context and question as inputs, this module outputs context embeddings and question embeddings by various approaches. Classical word representation methods like one-hot or word2vec, sometimes combined with other linguistic features, such as part-or-speech, name entity and question category, are usually utilized to represent semantic and syntactic information in the words. Moreover, contextualized word representations pre-trained by large corpus also show promising performance in encoding contextual information.

- Feature Extraction: After the Embedding module, embeddings of context and question are fed to the Feature Extraction module. To better understand the context and question, this module aims at extracting more contextual information. Some typical deep neural networks, such as Recurrent Neural Networks (RNNs) and Convolution Neural Networks (CNNs) are applied to further mine contextual features from context and question embeddings.

- Context-Question Interaction: The correlation between the context and question plays a significant role in predicting the answer. With such information, the machine is capable of finding out which parts in the context are more important to answering the question. So as to achieve that goal, attention mechanism, unidirectional or bidirectional, is wildly utilized in this module to emphasize parts of the context relevant to the query. In order to sufficiently extract their correlation, the interaction between the context and question is sometimes performs multiple hops which simulates the rereading process of human comprehension.

- Answer Prediction: Answer Prediction module is the last component of MRC systems, which outputs the final answer base on the whole information accumulated from previous modules. As MRC tasks can be categorized according to answer forms, this module is highly related to different tasks. For cloze tests, the output of this module is a word or an entity in the original context, while multiple choice task asks to select the right answer from candidate answers. In terms of span extraction, this module extracts a subsequence of the given context as the answer. Some generation techniques are utilized in this module for free answering task, as there is nearly no constraint on answer forms in that task.

4 Methods

Compared to traditional rule-based methods, deep learning techniques show their superiority in extracting contextual information, which is very important to MRC tasks. In this section, we firstly present various deep learning approaches utilized in different modules of MRC systems in Fig 4 and introduce some tricks applied to improve the performance of MRC systems.
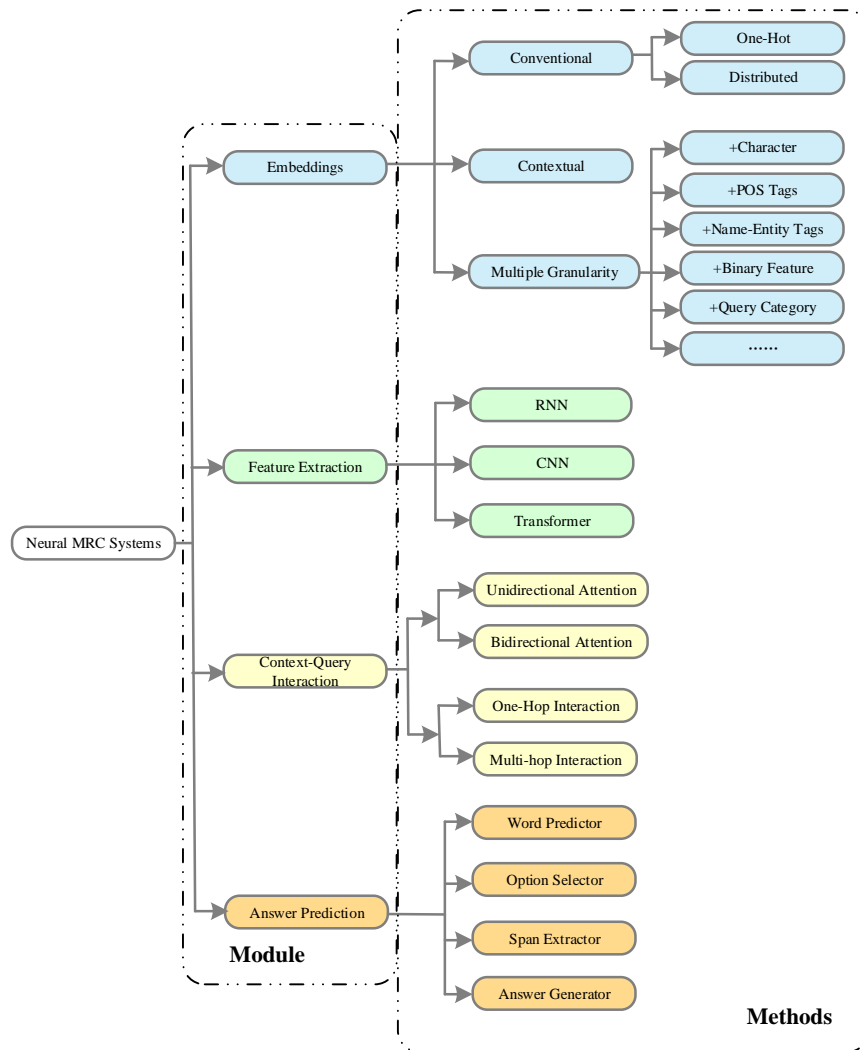
Figure 4: Typical techniques in neural MRC systems.

4.1 Embeddings

Embedding module is an essential part in MRC systems and usually placed at the beginning to encode inputs natural language words into fixed-length vectors, which the machine can understand and deal with. As Dhingra et al. [18] point out, the minor choices made in word representation can lead to substantial differences in the final performance of the reader. How to sufficiently encode the context and question is the pivot task in this module. In existing MRC models, word representation methods can be sorted into conventional word representation and pre-trained contextualized representation. To encode more abundant semantic and linguistic information, multiple granularity, which fuses word-level embeddings with character-level embeddings, part-of-speech, name entity, word frequency, question category and so on, is also applied to some MRC systems. In the following parts, we will give a detailed illustration.

(1) Conventional Word Representation

- One-Hot

This method [1] represents a word with a binary vector, whose size is same as the number of words in the dictionary. In such vectors, just one position is 1 corresponding to the word while the others are all 0. As a word representation approach at the early stage, it can encode words when vocabulary size is not very large. However, this representation is sparse and may suffer from the curse of dimensionality with the increase of vocabulary size. In addition, one-hot encoding can not represent relation among words. For instance, "apple" and "pear" belong to fruit category, but their word representations embedded by one-hot cannot show such relation.

- Distributed Word Representation

To address shortcomings of representations like one-hot, Rumelhart et al. [67] propose distributed word representation, which encodes words into continuous low-dimensional vectors. Closely-related words encoded by these methods are not far away from each other in vector space, which reveals correlation of words. Various techniques to generated distributed word representations have been introduced, among which the most popular ones are Word2Vec [47] and GloVe [58]. Besides success applications in a variety of NLP tasks like machine translation [11], sentiment analysis [50], vectors produced by these methods are also applied to a large number of MRC systems.

(2) Pre-Trained Contextualized Word Representation

Although distributed word representation can encode words in low dimensional space and reflect correlation between different words, they cannot efficiently mine contextual information. To be specific, vectors produced by distributed word representation for one word is constant regardless of different context. To address this problem, researchers introduce contextualized word representations, which are pre-trained with large corpus in advance and then directly utilized just as conventional word representation or fine-tuned according to specific tasks. This is a kind of transfer learning and has shown promising performance in a wide range of NLP tasks including machine reading comprehension. Even a simple neural network model can perform well in answer prediction with these pre-trained word representation approaches.

- CoVE

Inspired by successful case in computer vision, which transfers CNNs pre-trained on large supervised training corpus like ImageNet to other tasks, McCann et al. [45] try to bring beneficial of transfer learning to NLP tasks. They firstly train LSTM encoders of the sequence-to-sequence models on a large-scale English-to-German translation dataset and then transfer the outputs of encoder to other NLP tasks. As Machine Translation (MT) require the model to encode words in the context, the outputs of encoder can be regarded as context vectors (CoVE). To deal with MRC problems, McCann et al. concatenate the outputs of MT encoder with word embeddings pre-trained by GloVe to represent the context and question and feed them through the coattention and dynamic decoder implemented in DCN [94]. DCN with CoVe outperforms the original one on SQuAD dataset, which illustrates the contribution of contextualized word representations to downstream tasks. However, pre-training CoVE requires a great deal of parallel corpus. Its performance will degrade if the training corpus is not adequate.

- ELMo

Embeddings from Language Models (ELMo), proposed by Peters et al. [59], is another contextualized word representation. To get ELMo embeddings, they firstly pre-trained a bidirectional Language Model (biLM) with a large text corpus. Compared to CoVe, ELMo breaks the constraint of limited parallel corpus and can obtain richer word representations by collapsing outputs of all biLM layers into a single vector with a task specific weighting rather than just utilizing outputs of the top layer. Model evaluations illustrate that different levels of LSTM states can capture diverse syntactic and linguist information. When applying ELMo embeddings to MRC models, Peters et al. choose an improved version of Bi-DAF introduced by Clark and Gardner [13] as baseline and improve the state-of-the-art single model by 1.4% on SQuAD dataset. ELMo, which can easily be integrated to existing models, shows promising performance on various NLP tasks, but it is limited in a way by the insufficient feature extraction capability of LSTM.

- GPT

GPT [61], short for Generative Pre-Training, is a semi-supervised approach combining unsupervised pre-training and supervised fine-tuning. Representations pre-trained by this method can transfer to various NLP tasks with little adaptation. The basic component of GPT is a multi-layer Transformer[82] decoder which mainly use multi-head self-attention to train the language model

and allow to capture longer semantic structure compared to RNN-based models. After training, the pre-trained parameters are fine-tuned for specific downstream tasks. In terms of MRC problems like multiple choice, Radford et al. concatenate the context and question with each possible answer and process such sequences with Transformer networks. Finally, they produce an output distribution over possible answers to predict correct answer. GPT achieves improvements of 5.7% on RACE[36] dataset compared with state-of-the-art. Seeing the beneficial brought by contextualized word representations pre-trained on large-scale datasets, Radford et al. [62] propose GPT-2 later, which is pre-trained on larger corpus, WebText, with more than 1.5 billion parameters. Compared to the previous one, layers of Transformer architecture increase from 12 to 48. Moreover, single task training is substituted with multitask learning framework, which makes GPT-2 more generative. This improved version can show competitive performance even in zero-shot setting. However, Transformer architecture utilized in both GPT and GPT-2 is unidirectional (left-to-right), that cannot incorporate context from both directions. This may be the major shortcoming and limits its performance on downstream tasks.

- BERT

Considering the limitations of unidirectional architecture applied in previous pre-training models like GPT, Devlin et al. [17] propose a new one named BERT (Bidirectional Encoder Representation from Transformers) . With the masked language model (MLM) and next sentence prediction task, BERT is able to pre-train deep contextualized representations with bidirectional Transformer, encoding both left and right context to word representations. As Transformer architecture cannot extract sequential information, Devlin et al. add positional embeddings to encode position. Owing to bidirectional language model and Transformer architecture, BERT outperforms state-of-the-art models in eleven NLP tasks. In particular, for MRC tasks, BERT is so competitive that just utilizing BERT with simple answer prediction approaches can show promising performance. Despite of its outstanding performance, pre-training process of BERT is time and resource consuming which makes it nearly impossible to be pre-trained without abundant computational resources.

(3) Multiple Granularity

Word-level embeddings pre-trained by Word2Vec or GloVe cannot encode rich syntactic and linguistic information, such as part-of-speech, affixes and grammar, which may not be sufficient for deep machine understanding. In order to incorporate fine-grained semantic information to word representations, some researchers introduce approaches to encode the context and question at different levels of granularity.

- Character Embeddings

Character embeddings represent a word in character level. Compared to word-level representations, they are not only more suitable for modeling sub-word morphologies but also can alleviate out-of-vocabulary (OOV) problem. Seo et al. [70] firstly add character-level embeddings in their Bi-Daf model for the MRC task. They use Convolutional Neural Networks (CNNs) to obtain character-level embeddings. Each character in the word is embedded into a fixed-dimension vector, which is fed to CNNs as 1D inputs. After max-pooling the entire width, the outputs of CNNs are embeddings in character level. The concatenation of word-level embeddings and character-level embeddings are then fed to next module as inputs. In addition, character embeddings can also be encoded with bidirectional LSTMs [28, 90]. For each word, the outputs of last hidden state are considered as its character level representation. Besides, word-level embeddings and character-level embeddings can be combined dynamically with a fine-grained gating mechanism rather than simple concatenation to mitigate the imbalance between frequent words and infrequent words [97].

- Part-of-Speech Tags

Part-of-speech (POS) is a particular grammatical class of word, such as noun, adjective, verb. Labeling POS tags in NLP tasks can illustrate complex characteristic of word use and in turn contribute to disambiguation. To translate POS tags into fix-length vectors, they are regarded as variables, randomly initialized in the beginning and updated while training.

- Name-Entity Tags

Name entity, a concept in information retrieval, refers to a real-world object, such as persons, locations, organizations and so on, with a proper name. When asking about such objects, name entities are probable answer candidates. Thus, embedding name-entity tags of context words can

improve accuracy of answer prediction. The method to encode name-entity tags is similar to pos tags mentioned above.

- Binary Feature of Exact Match (EM)

This feature, which measures whether a context word is in the question, is firstly used in the conventional entity-centric model proposed by Chen et al. [7]. Later, some researchers utilize it in Embedding module to enrich word representations. The value of this binary is 1 if a context word can be exactly matched to one word in the query, otherwise its value is 0. More loosely, Chen et al. [9] use partial matching to measure the correlation between context words and question words. For instance, "teacher" can be partially matched with "teach".

- Query-Category

The types of questions (i.e. what, where, who, when, how) can usually provide clues for searching the answer. For instance, a question with "where" pays more attention to spatial information. Zhang et al. [102] introduce a method to model different question categories in the end-to-end training. They firstly obtain query types by counting the key word frequency. Then the question types information is encoded to one-hot vectors and stored in a table. For each query, they look up the table and use a feedforward neural network for projection. The query-category embeddings are often added into the query word embeddings.

Embeddings introduced above can be combined freely in Embedding module. Hu et al. [28] use word level, character level, pos tags, name-entity tags, binary feature of EM and query-categories embeddings in their Reinforced Mnemonic Reader to incorporate syntactic and linguistic information to word representations. Experiment results show that rich word representations contribute to deep understanding and improve answer prediction accuracy.

To sum up, word embeddings encoded by distributed word representation are the basic of this module. As more abundant representations with syntactic and linguistic information contribute to better performance, multiple granularity representations have gradually become prevalent. In terms of contextual word representation, they can improve performance dramatically, which can be utilized solely or combined with other representations.

4.2 Feature Extraction

Feature Extraction module is often placed after the Embedding layer to extract features of the context and question separately. It further pays attention to mining contextual information in sentence level based on various syntactic and linguistic information encoded by the Embedding module. Recurrent Neural Networks (RNNs), Convolution Neural Networks (CNNs) and Transformer architecture are applied in this module, and we will give an illustration in detail in this part.

(1) Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are popular models that have shown great promise for dealing with sequential information. RNNs are called *recurrent* as outputs in each time step are depended on the previous computations. RNN-based models have been widely utilized in various NLP tasks, such as machine translation, sequence tagging and question answering. Especially, Long Short-Term Memory (LSTM) [26] Networks and Gated Recurrent Units (GRU) [11], variants of RNNs, are much better at capturing long-term dependencies than vanilla ones are and can alleviate gradient explosion and vanishing problems. Since the preceding and following words play the same importance in understanding the current word, many researchers utilize bidirectional RNNs to encode the context and question embeddings in MRC systems. The context embeddings and question embeddings are denoted as x_p and x_q, respectively, and then we will illustrate how Feature Extraction module with bidirectional RNNs handles those embeddings and extracts sequential information.

In terms of questions, the feature extraction process with bidirectional RNNs can be sorted into two types: word-level and sentence-level.

In word-level encoding, feature extraction outputs for each question embedding x_{qj} at time step j can be denoted as follows:

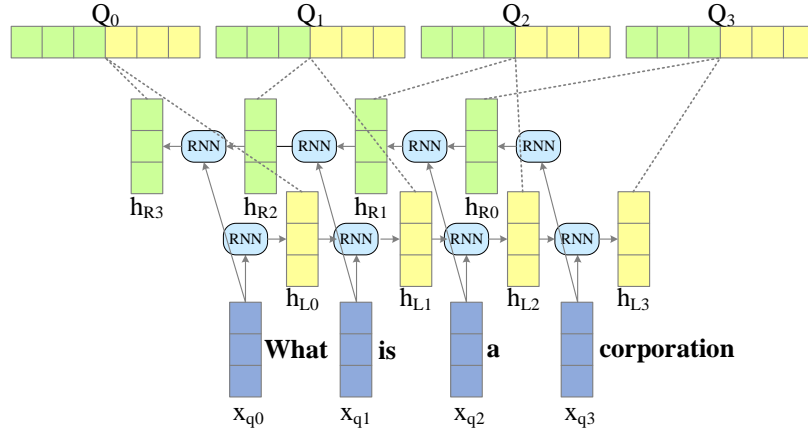$$Q_j = \overrightarrow{RNN}(x_{qj}) || \overleftarrow{RNN}(x_{qj}), \tag{11}$$

Figure 5: Word-level encoding for questions.

where $\overrightarrow{RNN}(q_{xj})$ and $\overleftarrow{RNN}(q_{xj})$ denotes forward and backward hidden states of bi-directional RNNs, respectively, and $||$ means the concatenation. This process is shown in Fig. 5 detailedly.

By contrast, sentence-level encoding method regards the question sentence as a whole. The feature extraction process can be denoted as:

$$Q = \overrightarrow{RNN}(x_{q|l|})||\overleftarrow{RNN}(x_{q0}), \tag{12}$$

where $|l|$ is the length of the question, $\overrightarrow{RNN}(x_{q|l|})$ and $\overleftarrow{RNN}(x_{q0})$ represent final forward and backward outputs of RNNs, respectively. To be more concrete, we demonstrate this sentence-level encoding process in Fig. 6
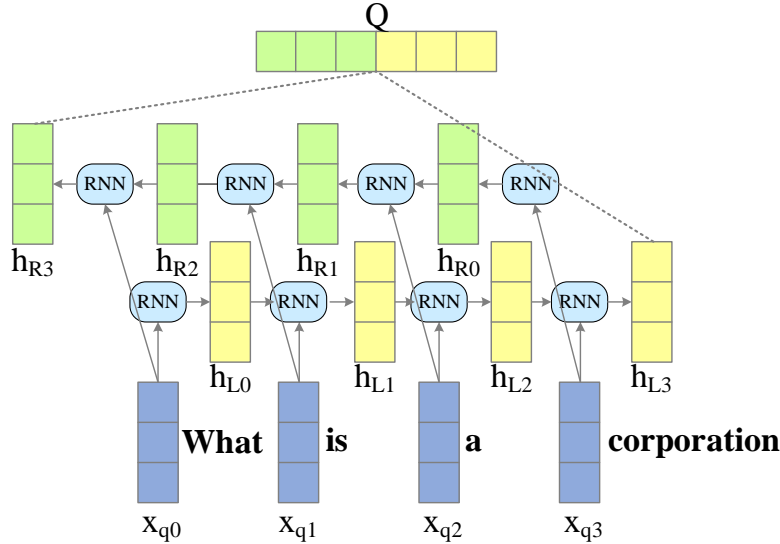

Figure 6: Sentence-level encoding for questions.

As the context in MRC tasks is usually long sequence, researchers just utilize word-level feature extraction method to encode sequential information of context. Similar to question encoding, feature

extraction process with bidirectional RNNs for the context embedding x_{ci} at time step i can be denoted as:

$$P_i = \overrightarrow{RNN}(x_{pi})||\overleftarrow{RNN}(x_{pi}). \tag{13}$$

Although RNNs are capable of modeling sequential information, their training process is time-consuming as they cannot be processed parallelly.

(2) Convolution Neural Networks

Convolution Neural Networks (CNNs) are wildly utilized in computer vision at first. When applied in NLP tasks later, one dimensional CNNs show their superiority in mining local contextual information with sliding windows. In CNNs, each convolution layer applies different scale of feature maps to extract local features in diverse window size. The outputs are then fed to pooling layers to reduce dimensionality but keep the most significant information to the greatest extent. Maximum and average operation to the results of each filter are common way to do pooling. Fig 7 presents how Feature Extraction module use CNNs to mine local contextual information of question.
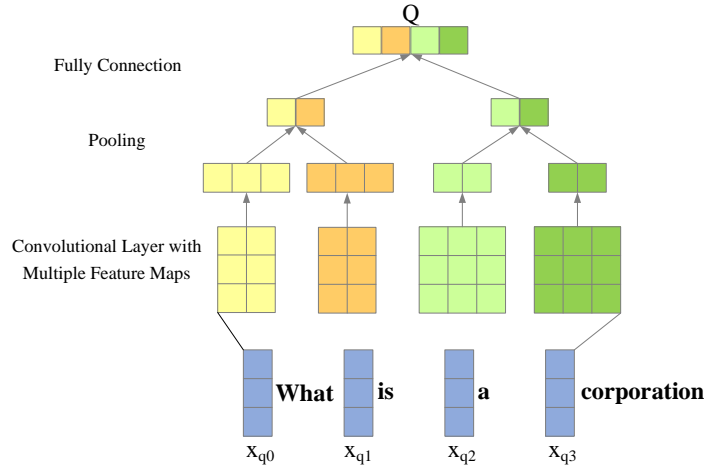

Figure 7: Utilizing CNNs to extract features of question.

As shown in Fig 7, given word embeddings of question $x_q \in \mathbb{R}^{|l| \times d}$, where $|l|$ represents the length of questions and d denotes the dimension of word embeddings, the convolution layer has two types of filters of sizes $f_t \times d(\forall t = 2, 3)$ with k output channels ($k = 2$ in the example presented in Fig 7). Each filter produces a feature map of shape $(|l| - t + 1) \times k$ (padding is 0 and stride is 1), which is pooled to generate a k-dimensional vector. The two k-dimensional vectors are concatenated to form a 2k-dimensional vector as representation Q.

Although both n-gram models and CNNs can focus on local features of the sentence, training parameters in n-gram models increase exponentially with the size of vocabulary being larger. By contrast, CNNs can extract local information in a more compact and efficient way regardless of the vocabulary size, for there is no need for CNNs to represent every n-gram in the vocabulary. In addition, CNNs can be trained parallelly, which are faster than RNNs. One major shortcoming of CNNs is that they can just extract local information, but are not capable of dealing with long sequence.

(3) Transformer

The Transformer, proposed by Vaswani et al. [82] in 2017, is a powerful neural network model that has shown promising performance in various NLP tasks [61, 17]. Different from RNNs-based or CNNs-based models, the Transformer is mainly based on attention mechanism with neither recurrence nor convolution. Owing to multi-head self-attention, this simple architecture not only excels in alignment but also is parallelized. Compared to RNNs, the Transformer requires less time to train, while it pays more attention to global dependencies in contrast with CNNs. However, without

recurrence and convolution, the model cannot make use of the order of the sequence. To incorporate positional information, Vaswani et al. add position encoding computed by sine and cosine functions. The sum of positional embeddings and word embeddings are fed to the Transformer as inputs. Fig. 8 present simple architecture of the Transformer. In practice, models usually stack several blocks with multi-head self-attention and feed-forward network.
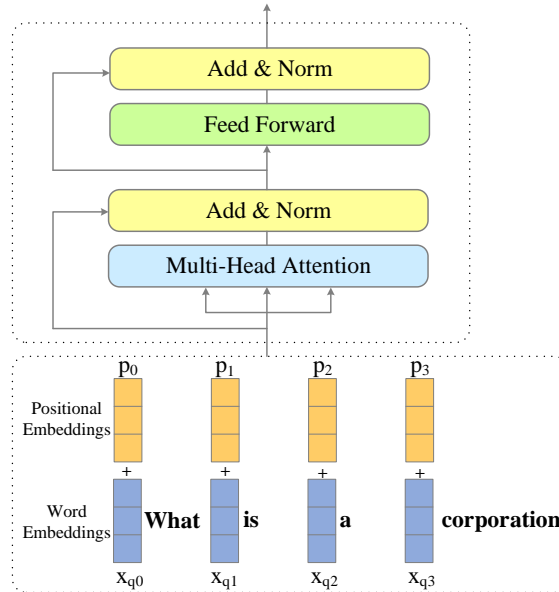
Figure 8: Utilizing the Transformer to extract features of question.

QANet, introduced by Yu et al. [99], is a representative MRC models with the Transformer. The basic encoder block of QANet is a novel architecture, which combines multi-head self-attention defined in the Transformer with convolutions. The experiment results show that QANet achieves same accuracy on SQuAD as prevalent recurrent models with much faster training and inference speed.

In general, most of MRC systems utilize RNNs in Feature Extraction Module because of their superiority in handling sequential information. Besides, in order to accelerate the training process, some researchers substitute RNNs with CNNs or the Transformer. CNNs are highly parallelized and can obtain rich local information with feature maps in different sizes. The Transformer can mitigate the side-effect of long dependency problem and improve computational efficiency.

4.3 Context-Question Interaction

By extracting the correlation between the context and question, models are capable of finding out evidence for answer prediction. Inspired by Hu et al. [28], existing works can be divided into two kinds according to how models extract correlation, one-hop and multi-hop interaction. No matter what kind of interaction MRC models utilize, attention mechanism plays a critical role in emphasizing which parts of contexts are more important to answer the questions.

Derived from human intuition, attention mechanism is firstly adapted to machine translation and shows promising performance on automatic token alignment [2, 43]. Later, as a simple and effective method that can be used to encode sequence data with its importance, it has attained significant improvement in various tasks in natural language processing including text summarization [68], sentiment classification [89], semantic parsing [10], etc. In terms of machine reading comprehension, attention mechanism can be categorized into unidirectional attention and bidirectional attention according to whether it is utilized unidirectionally or bidirectionally. In the following part, we will firstly introduce methods categorized by the use of attention mechanism, followed by the illustration of one-hop and multi-hop interaction, respectively.

(1a) Unidirectional Attention

Unidirectional attention flow is usually from query to context, highlighting the most relevant parts of the context according to the question. It is believed that if the context word is the more similar to the question, it is more likely to be the answer word. As shown in Fig. 9, the similarity of each context semantic embedding P_i and the whole question sentence representations Q (by sentence-level encoding introduced in 4.2) is calculated by $S_i = f(P_i, Q)$, where $f(\cdot)$ represents the function which can measure the similarity. After normalized by the softmax function in Equation 14, attention weight α_i for each context word is obtained, with which the MRC systems can finally predict the answer.

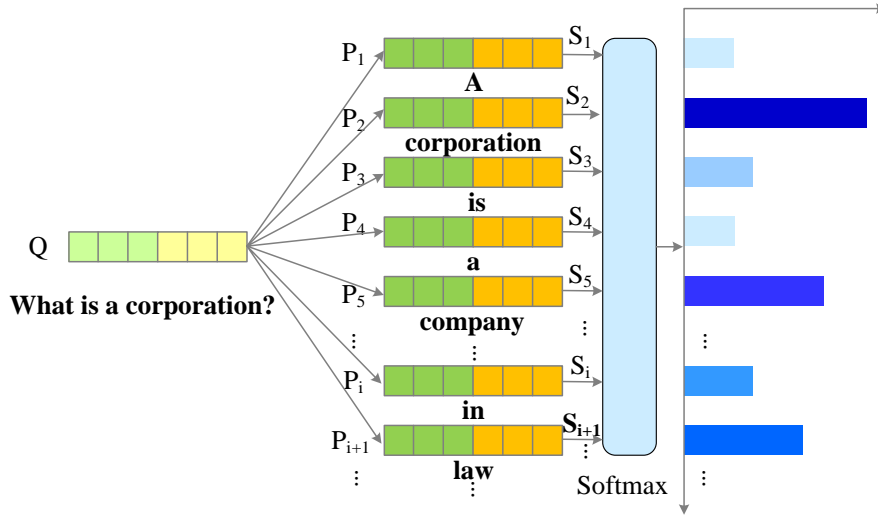$$\alpha_i = \frac{\exp S_i}{\sum_j \exp S_j}. \tag{14}$$

Figure 9: Using unidirectional attention to mine correlation between the context and question.

The choice of function $f(\cdot)$ differs from different models.

In the Attentive Reader, proposed by Hermann et al. [24], a tanh layer is used to computed the relevance between the context and question as follows:

$$S_i = \tanh(W_P P_i + W_Q Q), \tag{15}$$

where W_P and W_Q are trainable parameters.

Following the work of Hermann et al., Chen et al. [6] substitute bilinear term for tanh function as Equation 16:

$$S_i = Q^T W_s P_i, \tag{16}$$

which makes the model simpler and more effective than the Attentive Reader.

Unidirectional attention mechanism can highlight the most important context words to answering the question. However, this method fails to pay attention to question words which are also pivotal for answer prediction. Hence, unidirectional attention flow is insufficient for extracting mutual information between the context and query.

(1b) Bidirectional Attention

Seeing the limitations of unidirectional attention mechanism, some researchers introduce bidirectional attention flows, which not only compute query-to-context attention but also the reverse one, context-to-query attention. This method, which makes a mutual look from both directions, can benefit from the interaction between the context and query and provide complementary information for each other.

Fig. 10 presents the process of computing bidirectional attention. Firstly, the pair-wise matching matrix $M(i, j)$ is obtained by computing the matching scores between each context semantic embedding P_i and question semantic embedding Q_j (by word-level encoding introduced in 4.2). Then the outputs of column-wise softmax function can be regarded as query-to-context attention weight α while the context-to-query attention β are calculated by row-wise softmax function.
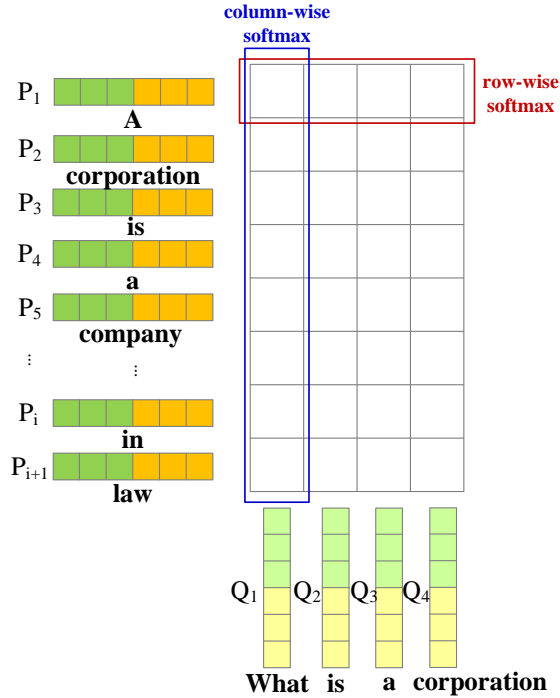

Figure 10: Using bidirectional attention to mine correlation between the context and question.

The Attention-over-attention Reader (AoA Reader) model, the Dynamic Coattention Network (DCN) and the Bi-directional Attention Flow (BiDAF) network are representative MRC models with bidirectional attention.

In the AoA Reader, Cui et al. [14] compute the dot product between each context embedding and query embedding to obtain the similarity matching matrix $M(i, j)$. The query-to-context and context-to-query attention are calculated as presented in Fig 10. To combine these two attention together, different from previous work in CAS Reader [15] that use naive heuristics, such as sum and average, over the query-to-context attention, Cui et al. introduce attended attention, computed by dot product of α and average result of β, which is later utilized to predict the answer.

In order to attend to the question and document simultaneously, Xiong et al. [94] fuse the two directional attention as follows:

$$C = \alpha[Q; \beta P], \tag{17}$$

where C can be regarded as the coattention representations which contains attention information of both context and question. Based on DCN, Xiong et al. [95] later introduce its extension DCN+, using residual connections to merge co-attention outputs together to encode richer information to the input sequences. Compared to the work of AoA Reader, Xiong et al. further calculate context

representations with two directional attentive information, rather than directly utilizing attention weights for answer prediction, which can better extract the correlation between the context and question.

Different from the AoA Reader and DCN, which directly summarize the output of two directional attention flows, Seo et al. [70] let attentive vectors flow into another RNN layer to encode query-aware context representations, which can reduce information loss caused by early summarization. To be concrete, after obtaining both query-to-context attention weight α and context-to-query attention weight β, Seo et al. compute attended context vector $\widetilde{P}$ and attended query vector $\widetilde{Q}$ as follows:

$$
\begin{aligned}
\widetilde{P} &= \sum_i \alpha P_i, \\
\widetilde{Q} &= \sum_j \beta Q_j.
\end{aligned}
\tag{18}
$$

Then the context embeddings and attention vectors are combined together by a simple concatenation:

$$
G = [P; \widetilde{Q}; P \circ \widetilde{Q}; P \circ \widetilde{P}],
\tag{19}
$$

where $\circ$ is element-wise multiplication and G can be regarded as query-aware context representations, which are later fed to bi-directional LSTM to be further encoded.

To sum up, MRC systems at the early stage usually utilize unidirectional attention mechanism, especially query-to-context attention, to highlight which part of the context is more important to answer the question. However, query-to-context attention is not sufficient to extract the mutual information between the context and query. Later, bidirectional attention is wildly applied to overcome the shortcoming of the unidirectional one, which can benefit from context-query correlation and output attentive representations with the fusion of the context and question information.

(2a) One-Hop Interaction

One-Hop interaction is a shallow architecture, where the interaction between the context and question is computed only once. At the early time, context-query interaction is such one-hop architecture in many MRC systems, for example, the Attentive Reader [24], the Attention Sum Reader [33], the AoA Reader [14] and so on. Although this method can do well in tackling simple cloze tests, when the question requires reasoning over multiple sentences in the context, it is hard for this one-hop interaction approach to predict the right answer.

(2b) Multi-Hop Interaction

In contrast to one-hop interaction, multi-hop interaction is much more complex and try to mimic the rereading phenomenon of human with the memory of the context and question. In the process of interaction, whether information of previous state can be efficiently stored or not directly effects the performance of next interaction.

There are mainly three methods to perform multi-hop interaction:

The first one calculates the similarity between the context and question based on the previous attentive representations of context. In the Impatient Reader model, proposed by Hermann et al. [24], the query-aware context representations are dynamically updated by this method as each query token is read. This stimulates the process that human reread the given context with the question information.

The second one introduces external memory slots to store previous memories. The representative models utilizing this method is Memory Networks, proposed by Weston et al. [92], which can explicitly store long-term memories and also have an easy access to reading memories. With such mechanism, MRC models can understand the context and question more deeply by multiple turns of interaction. After given the context as input, memory mechanism stores the context information into memory slots and then updates them dynamically. The process of answering is to find out the most relevant memory to the question and turn it into answer representations as required. Although this method can overcome the shortcoming of insufficient memory, the network is hard to be trained via back-propagation. To address this problem, Sukhbaatar et al. [74] later introduce an end-to-end

version of memory networks. Compared to the previous one, explicit memory storage is embedded with continuous representations. Moreover, the process of reading and updating memories is modeled by neural networks. This extension of memory networks can reduce supervision during training and applicable to more tasks.

The characteristic of memory networks that it can update memories multiple hops makes it popular in MRC systems. Pan et al. [54] propose the MEMEN model, which stores question-aware context representations, context-aware question representations and candidate answer representations into memory slots and updates them dynamically. Similarly, Yu et al. [100] use external memory slots to store question-aware context representations and update memories with bi-directional GRUs.

The third one takes advantage of the recurrence feature of RNNs, using hidden state to store the previous interaction information. Wang & Jiang [85] perform multiple interaction by using match-LSTM architecture recurrently. This model is originally proposed for textual entailment, when introduced to MRC, it can simulate the process of reading passages with question information. Firstly, Wang & Jiang use standard attention mechanism to obtain attentive weights of each context token to the question. After calculating the dot product of question tokens and attentive weights, the model concatenates it with the context token and feeds it to match-LSTM to get query-aware context representations. Similarly, this process is done in the reverse direction in order to fully encode contextual information. Finally, the outputs of match-LSTM in two directions are concatenated together and are later fed to answer prediction module. In addition, R-Net [88], IA Reader [72] and Smarnet [8] also utilize RNNs to update the query-aware context representations to perform multi-hop interaction.

Some early work treats each context and query word equally when mining their correlation. However, the most important part should be given more attention for efficient context-query interaction. Gate mechanism, which can control the amount of mutual information between the context and question, is a key component in multi-hop interaction.

In the Gated-Attention (GA) Reader, Dhingra et al. [19] utilize gate mechanism to decide how question information affects the focus on the context words when updating the context representations. The gate attention mechanism is performed by an element-wise multiplication between query embeddings and intermediate representations of context more than one time.

Different from the GA Reader, both context and question representations are updated in the Iterative Alternating attention mechanism [72]. Question representations are updated with previous search state while context representations are refined not only with previous reasoning information but also currently updated query. Then the gate mechanism, which is performed by feed-forward network, is applied to determine the degree of matching between the context and query. This mechanism is capable of extracting evidence from the context and question alternantly.

In the Smarnet model, Chen et al. [8] not only use gate mechanism to control the question influence on the context, but also introduce another gate mechanism to refine query representations with the knowledge of context. The combination of these two gated-attention mechanisms implements the alternant reading between the context and question with mutual information.

Previous models ignore that context words have different importance to answer particular questions. To address this problem, Wang et al. [88] introduce the gate mechanism to filter out the insignificant parts in the context and emphasize the most relevant ones to the question in their R-NET model. This model can be regard as a variant of attention-based recurrent networks. Compared to match-LSTM [85], it introduces additional gate mechanism based on the current context representations and context-aware question representations. Moreover, as RNNs-based models cannot deal well with long documents because of insufficient memories, Wang et al. add self attention to the context itself. This mechanism can dynamically refine context representations based on mutual information from the whole context and question.

In conclusion, one-hop interaction may fail to comprehensively understand the mutual question-context information. By contrast, multiple-hop interaction with the memory of previous context and question is capable of deeply extracting correlation and aggregating evidence for answer prediction.

4.4 Answer Prediction

This module is always at the last of MRC systems which gives answers to questions according to the original context. The implementation of answer prediction is highly task-specific. As MRC tasks are categorized into cloze tests, multiple choice, span extraction and free answering in section 2.1, there are four kinds of answer prediction methods: word predictor, option selector, span extractor and answer generator. In this part, we will give an illustration in detail.

(1) Word Predictor

The cloze tests are required to fill in the blank with the missing word or entity. Namely, it is asked to find out a word or an entity from the given context as the answer. At the early work like the Attentive Reader [24], the combination of query-aware context and question are reflected in the vocabulary space to search for the right answer word. Chen et al. [6] directly utilize the query-aware context representations to match the candidate answer, which simplifies the prediction process and improves the performance.

Above method employs attentive context representations to select the right answer word, but it cannot ensure that answers are in the context, which is not satisfied with the requirements of cloze tests. A related example is shown below (Kadle et al. [33]).

> *Context:*   *A UFO was observed above our city in January and again in March.*
> *Question:*   *An observer has spotted a UFO in _____.*

In the condition that both *January* and *March* can be the right answer, methods utilized by Hermann et al. [24] and Chen et al. [6], which reflect attentive context representations into the whole vocabulary space, would give an answer similar to these two words, maybe *February* because of features of distributed word representation pre-trained by Word2Vec.

To overcome the problem that predicted answer may not in the context, Kadlec et al. [33] propose the Attention Sum (AS) Reader model inspired by the pointer networks. Pointer networks, introduced by Vinyals et al. [83], is adapted to the tasks whose outputs can only be selected from inputs at first and can well satisfy the requirement of cloze tests. In the AS Reader, Kadlec et al. do not compute the attentive representations, instead directly utilize attention weights to predict the answer. The attention results of the same word are added together and the one with maximum value is selected as the answer. This method is simple but quite efficient for cloze tests.

(2) Option Selector

To tackle the multiple choice task, the model should select the right answer from candidate answer options. The common way is to measure the similarity between attentive context representations and candidate answer representations and the most similar candidate is chosen as the right answer.

Chaturvedi et al. [4] utilize CNNs to encode the question-option tuples and relevant context sentences. Then the correlation between them is measured by the cosine similarity. The most relevant option is selected as the answer. Zhu et al. [104] introduce the information of options to contribute to extracting the interaction between the context and question. In answer prediction module, they use bilinear function to score each option according to the attentive information. The one with highest score is the predicted answer. In Convolutional Spatial Attention model, Chen et al. [9] calculate similarity among question-aware candidate representations, context-aware representations and self-attended question representations with dot product to fully extract correlation among the context, question and options. The diverse similarity are concatenated together and then fed to CNNs with different kernel sizes. The outputs of CNNs are regarded as feature vectors and fed to fully-connected layer to calculated a score for each candidate. Finally, the correct answer is the one with highest score.

(3) Span Extractor

The span extraction task can be regarded as the extension of cloze tests, which requires to extract a subsequence from the context rather than a single word. As word predictor methods utilized in models for cloze tests can only extract one context token, they cannot directly be applied to the span extraction task. Also inspired by pointer networks [83], Wang & Jiang [85] propose two different models, the Sequence Model and the Boundary Model to overcome the shortcomings of

word prediction approaches. Outputs of the Sequence Model are positions where answer tokens appear in the original context. The process of answer prediction is similar to decoding of sequence-to-sequence models, which selects tokens with highest probability successively until stop answer generating token. Answers obtained by these methods are treated as a sequence of tokens from input context which might not be consecutive span and cannot ensure to be a subsequence of original context. The Boundary Model can well handle this problem, which only predict the start and the end position of the answer. The Boundary Model is much simpler and shows better performance on SQuAD. Then it is wildly used in other MRC models as preferred alternative for subsequence extraction.

Considering that there is more than one plausible answer span in the original context, but the boundary model would extract incorrect answer with local maxima, Xiong et al. [94] proposed a dynamic pointing decoder to select an answer span by multiple iterations. This method utilizes LSTM to estimate the start and end position based on representations corresponding to last state answer prediction. To compute start and end score of context tokens, Xiong et al. propose Highway Maxout Networks (HMN) with Maxout Networks [22] and Highway Networks [73], which require different models according to various question types and context topics.

(4) Answer Generator

With the appearance of free answering tasks, answers are no longer limited to sub-span of the original context, instead need to be synthesized from both the context and question. Specifically, expression of answers may different from the evidence snippet in the given context or answers may from multiple evidence even in different passages. Answer forms of the free answering task have the least limits, but in turn this task propose high requirements for Answer Prediction module. In order to deal with the challenge, some generation approaches are introduced to generate flexible answers.

S-Net, proposed by Tan et al. [78], introduces the answer generation module to satisfy the requirement of free answering tasks, whose answers are not limited to the original context. It follows the "extraction and then synthesis" process. The extraction module is a variant of R-Net [88] while the generation module is a sequence-to-sequence architecture. To be concrete, for the encoder, bidirectional GRU is utilized to produce context and question representations. Especially, start and end positions of evidence snippets predicted by span extraction module are added to context representations as additional features. In terms of the decoder, the state of GRU is updated by the previous context word representations and attentive intermediate information. After the softmax function, the output of the decoder is the synthetic answer.

The introduction of the generation module successfully makes up for the deficiency of the extraction module and generates more flexible answers. However, answers generated by existing generation approaches may suffer from syntax errors and illogical problems. Hence, generation and extraction methods are usually utilized together to provide complementary information for each other. For example, in the S-Net, extraction module firstly labels approximate boundary of the answer span while generation module generates answers not limited to the original context based on that. Generation approaches are not very common in existing MRC systems, as the extraction methods have already performed well enough in most cases.

4.5 Other Tricks

(1) Reinforcement Learning

As can be seen from above introduction, most MRC models only apply maximum-likelihood estimation in training process. However, there is a disconnection between optimization objective with evaluation metrics. As a result, candidate answers which exactly match the ground-truth or have word overlap with the ground-truth but do not locate at the labeled position would be ignored by such models. In addition, when answer span is too long or with fuzzy boundary, models would also fail to extract the correct answer. For MRC evaluation metrics like exact match (EM), F1 are not differentiable, some researchers introduce reinforcement learning to training process. Xiong et al. [95] and Hu et al. [28] both utilize F1 score as the reward function and treat maximum-likelihood estimation and reinforcement learning as a multi-task learning problem. This method can take both textual similarity and position information into consideration.

Reinforcement Learning can also be used to determine whether to stop the interaction process. Multi-hop interaction methods introduced above have a pre-defined number of hops in interaction. However, when people answering the question, they stop reading if there is adequate evidence for giving the answer. The termination state is highly related to the complexity of the given context and question. With the motivation of stopping interaction dynamically according to the context and question, Shen et al. [71] introduce a termination state to their ReasonNets. If the value of this state equals 1, the model stops interaction and feeds the evidence to answer prediction module to give the answer, otherwise ReasonNets continues to interaction by computing the similarity between the intermediate state and input context and query. As the termination state is discrete and back-propagation approach cannot be used directly while training, reinforcement learning is applied to train the model by maximizing the instance-dependent expect reward.

In a word, reinforcement learning can be regarded as an improved approach in MRC systems which is capable of not only reducing the gap between optimization objection and evaluation metrics but also determining whether to stop reasoning dynamically. With reinforcement learning, the model can be trained and refine better answers even if some states are discrete.

(2) Answer Ranker

To verify whether the predicted answer is right or not, some researchers introduce answer ranker module. The common process of the ranker is that some candidate answers are firstly extracted and the one with highest rank score is the right answer.

EpiReader [81] combines pointer methods with the ranker. Trischler et al. firstly extract answer candidates using the approach similar to the AS Reader [33], selecting some answer spans with highest attention sum score. Then EpiReader feeds those candidates to the Reasoner component, which inserts candidates to the question sequence at placeholder location and computes their probability to be the right answers. The one with the highest probability is selected as the correct answer.

To extract candidates with variable lengths, Yu et al. [101] propose two approaches. In the first one, they capture the part-of-speech (POS) patterns of answers in the training set and choose subsequences in the given passage which can match such patterns as candidates. The other way enumerates all possible answer span within a fixed length from the context. After obtaining answer candidates, Yu et al. compute their similarity with question representations and choose the most similar one as the answer.

With the ranker module, the accuracy of answer prediction can be improved in a way. These methods also inspire some researchers to detect unanswerable questions later.

(3) Sentence Selector

In practice, if the MRC model is given a long document, it is time-consuming to understand the full context to answer the question. However, finding the most relevant sentences to the questions in advance is a possible way to accelerate the sequent training process. With this motivation, Min et al. [49] propose a sentence selector to find out the minimal set of sentences needed to answer the question. The architecture of the sentence selector is sequence-to-sequence, which contains an encoder to compute sentence encodings and question encodings and a decoder to calculate score for each sentence by measuring the similarity between sentence and question. If the score is higher than the pre-defined threshold, the sentence is selected to be fed to the MRC systems. By this way, the number of selected sentences is dynamic according to different questions.

MRC systems with sentence selector are capable of reducing training and inference time with equivalent or better performance compared to ones without sentence selector.

5 New Trends

With neural network models surpassing human performance on the representative MRC dataset, SQuAD, it seems that machine reading comprehension techniques have made great strides. However, due to the limitations of MRC tasks, there is still a long way to go before the machine truly understands text. To make MRC tasks much closer to real-world application, a lot of new trends spring up and we will give detailed introduction in this section.

5.1 Knowledge-Based Machine Reading Comprehension

In MRC, it is required to answer questions with implicit knowledge in the given context. Datasets like the MCTest choose passages from specific corpus (fiction stories, children's books, etc.) to avoid introducing external knowledge. However, those human-generated questions are usually too simple when compared to ones in real world application. In the process of human reading comprehension, we may utilize world knowledge when the question cannot be answered simply by the knowledge in the context. The external knowledge is so significant that is believed as the biggest gap between MRC and human reading comprehension. As a result, interests in introducing world knowledge to MRC have surged in research community and knowledge-based machine reading comprehension (KBMRC) comes into being. KBMRC differs from MRC mainly in inputs. In MRC, the inputs are sequence of the context and question. However, besides that, additional related knowledge extracted from knowledge base is necessary in KBMRC. In a word, KBMRC can be regarded as augmented MRC with external knowledge and it can be formulated as follows:

KBMRC

Given the context C, question Q and external knowledge K, the task requires to predict the right answer a by maximizing the conditional probability $P(a|C, Q, K)$.

There are some KBMRC datasets, in which world knowledge is a necessity to answer some questions. MCScripts [53] is a dataset about human daily activities, such as eating in a restaurant and taking a bus, where answering some questions asks for commonsense knowledge beyond the given context. As shown in Table 3, the answer to *What was used to dig the hole?* cannot be found in the given context. However, it is known to us as the commonsense knowledge that human always digs the hole with *a shovel* rather than *bare hands*.

Table 3: Some Examples in KBMRC

MCScripts	
Context:	I wanted to plant a tree. I went to the home and garden store and picked a nice oak. Afterwards, I planted it in my garden.
Question 1:	What was used to dig the hole?
Candidate Answers:	A. a shovel B. his bare hands
Question 2:	When did he plant the tree?
Candidate Answers:	A. after watering it B. after taking it home

The key challenges in KBMRC are listed below:

- Relevant External Knowledge Retrieval

    There are various knowledge stored in knowledge base and entities may be misleading sometimes because of polysemy, e.g., "apple" can refer to a fruit or an incorporation. How to extract knowledge closely related to the context and question determines the performance of knowledge-based answer prediction.

- External Knowledge Integration

    Different from text in the context and questions, knowledge in external knowledge base has its unique structure. How to encode such knowledge and integrate it with the representations of the context and questions remains an ongoing research challenge.

Some researchers have tried to address above challenges in KBMRC. To make the model take advantage of the external knowledge, Long et al. [42] propose a new task, rare entity prediction, which requires to predict the missing name entity and is similar to cloze tests. However, name entities removed from the context cannot be predicted correctly only based on the original context. This task provides additional entity description extracted from knowledge base like Freebase as external knowledge to help entity prediction. While incorporating external knowledge, Yang & Mitchell [96] consider the relevance between the knowledge and context to avoid that irrespective

external knowledge misleads the answer prediction. They design the attention mechanism with sentinel to determine whether to incorporate external knowledge or not and which knowledge should be adopted. Both Mihaylov & Frank [46] and Sun et al. [76] utilize Key-Value Memory Networks [48] to find out relevant external knowledge. All possible related knowledge is firstly selected from knowledge base and stored in memory slots as key-value pairs. Then keys are used to match with the query while corresponding values are weighted summed together to generate relevant knowledge representations. Wang & Jiang [84] propose a data enrichment method with semantic relations in WordNet, a lexical database for English. For each word in the context and question, they try to find out the positions of passage words, which have directly or indirectly semantic relations to that. This position information is regarded as external knowledge and fed to MRC models to assist answer prediction.

In conclusion, KBMRC breaks through the limitation that the scope of knowledge required to answer questions is restricted to the given context. Hence, this task, beneficial from the external world knowledge, can mitigate the gap between machine comprehension and human understanding to some extent. However, the performance of KBMRC systems is highly related to the quality of knowledge base. Efforts for disambiguation is required when extracting related external knowledge from automated or semi-automated generated knowledge base as entities with same name or alias may mislead the models. Moreover, knowledge stored in knowledge base is usually sparse. If related knowledge cannot be found directly, incorporating external knowledge calls for further inference.

5.2 Unanswerable Questions

There is a latent hypothesis behind MRC tasks that correct answers always exist in the given context. However, it is inconformity to the real world application. The range of knowledge covered in the passage is limited, thus some questions inevitably have no answers according to the given context. A mature MRC system should distinguish those unanswerable questions. The definition of this new task is shown below:

MRC with Unanswerable Questions

Given the context C and question Q, the machine firstly determines whether Q can be answered or not based on the given context C. If the question is impossible to be answered, the model marks it as unanswerable and abstain from answering, otherwise predicts the right answer a by maximizing the conditional probability $P(a|C,Q)$.

SQuAD 2.0 [63] is a representative MRC dataset with unanswerable questions. Based on the previous version released in 2016, SQuAD 2.0 has more than 50,000 unanswerable questions created by crowd-workers. Those questions, impossible to be answered based on the context alone, are challenging for they are relevant to the given context and there are plausible answer span whose type matches answer type the question requires. To perform well on SQuAD 2.0, a model not only gives correct answers to answerable questions, but also detects which questions have no answers. An example of unanswerable question in SQuAD 2.0 is presented in Table 4. In the context, keywords *1937 treaty* exist and *Bald Eagle Protection Act* is the name of the treaty in 1940, not in 1937, which is very puzzling.

Table 4: Unanswerable question example in SQuAD 2.0

SQuAD 2.0	
Context:	Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treatyprohibiting the hunting of right and gray whales, and the Bald Eagle Protection Act of 1940. These later laws had a low cost to society -the species were relatively rare -and little opposition was raised.
Question:	What was the name of the 1937 treaty
Plausible Answer:	Bald Eagle Protection Act

With unanswerable questions, there are another two challenges in this new task, compared to MRC:

- Unanswerable Question Detection

The model should know what they don't know. After comprehending the question and reasoning among the passage, the MRC models should judge which questions are impossible to be answered just based on the given context and mark them as unanswerable.

- Plausible Answer Discrimination

To avoid the impact of fake answers like the example presented in Table 4, the MRC model is required to verify the predicted answers and tell plausible answers from correct ones.

For the above two challenges, methods applied to tackle the problems in MRC with unanswerable questions can be categorized into two sorts:

To indicate no answer cases, one approach employs a shared-normalization operation between no-answer score and answer span score. Levy et al. [38] add an extra trainable bias to the confidence score of start and end position and apply softmax to the new score to obtain the probability distributions of no answer. If this probability is higher than that of the best span, it denotes the question is unanswerable, otherwise outputs the answer span. In addition, they also propose another method which sets a global confidence threshold, if the predicted answer confidence is below the threshold, the model labels the question as unanswerable. Although this approach can detect unanswerable questions, it cannot guarantee that predicted answers are correct to the question. The other methods introduce no-answer option by padding. Tan et al. [79] add a padding position for the original passage to determine whether the question is answerable. When the model predicts that position, it refuses to give an answer.

Researchers also pay much attention to the legitimacy of answers and introduce answer verification to discriminate plausible answers. For unanswerable question detection, Hu et al. [29] propose two auxiliary loss, Independent Span Loss to predict plausible answers regardless of the answerability of the question and Independent No-Answer Loss which alleviates the confliction between plausible answer extraction and no-answer detection tasks . In terms of answer verification, they introduce three methods. The first one, sequential architecture treats the question, answer, context sentence containing candidate answers as a whole sequence, and input that to the Finetuned Transformer model to predict the no-answer probability. The second one is interactive architecture, which calculates the correlation between question and answer sentence in the context to classify whether the question is answerable or not. The third one integrates above two approaches together by concatenating the outputs of two models as a joint representations and this hybrid architecture can yield better performance.

Different from above pipeline structure, Sun et al. [75] utilize multi-task learning to jointly train answer prediction, no answer detection and answer validation. What distinguishes their work is a universal node encoding passage and question information together, which is then integrated with question representations and answer position aware passage representations. After being passed through the linear classification layer, the fused representations can be utilized to determine whether the questions are answerable or not.

Just as the Chinese saying goes, *To know what it is that you know, and to know what it is that you do not know, that is wisdom.* The detection of unanswerable questions requires deep understanding of text and asks for more robust MRC models, making MRC much closer to real world application.

5.3 Multi-Passage Machine Reading Comprehension

In MRC tasks, the relevant passages are pre-identified, which contradicts to question answering process of human. People usually ask a question at first and then search for all possibly related passages where they find evidence to give the answer. To overcome this shortcoming, Chen et al. [7] extend MRC to machine reading at scale, more wildly called as multi-passage machine reading comprehension, which does not give one relevant passage for each question unlike tradition one. This extension can be applied to tackle open domain question answering tasks based on large corpus of unstructured text. With its appearance, some multi-passage MRC task-specific datasets have been released, such as MS MARCO [51], TriviaQA [32], SearchQA [21], Dureader [23], QUASAR [20].

In contrast to MRC, the definition of multi-passage MRC tasks changes to:

> **Multi-Passage Machine Reading Comprehension**
>
> Given a collection of m documents $\mathcal{D} = \{D_1, D_2, \cdots, D_m\}$ and the question Q, the multi-passage MRC task asks to give the right answer a to question Q according to documents $\mathcal{D}$ by maximizing the conditional probability $P(a|\mathcal{D}, Q)$.

Compared to MRC tasks, multi-passage MRC is far more challenging. For instance, although the DrQA model [7] achieves the exact match accuracy of 69.5 on SQuAD, when applied to open domain setting (using the whole Wikipedia corpus to answer the question), its performance drops dramatically. The unique features of multi-passage MRC listed below are the main reasons for the degradation:

- Massive Document Corpus

    This is the most prominent feature of multi-passage MRC, which makes it distinct from MRC given one related passage. Under this circumstance, whether a model can retrieve the most relevant documents from corpus fast and correctly or not decides the final performance of question answering.

- Noisy Document Retrieval

    Multi-passage MRC can be regarded as a distantly supervised open domain question answering task, which may suffer from noise issues. Sometimes the model may retrieve noisy document which contains the right answer span but have no relation with the question. This noise will mislead the understanding of the context.

- No Answer

    When the retrieval component does not perform well, there would be no answers in the document. If the answer extraction module ignores that, it outputs an answer even it is incorrect, which will lead to performance degradation.

- Multiple Answers

    In open domain setting, multiple answers for a single question is common. For example, when asking about *Who is the president of the United States*, both *Obama* and *Trump* are possible answers, but which one is the right answer requires reasoning based on the context.

- Evidence Aggregation

    In terms of some complicated questions, evidence snippets appear in different parts of one document or even in different documents. To answer such questions correctly, a multi-passage MRC model need to aggregate those evidence together. More documents mean more information, which contributes to more complete answers.

To address multi-passage MRC problems, one method follows the pipeline of "retrieve then read". To be more concrete, the retrieval component firstly returns several relevant documents, which are then proposed by the reader to give the answer. DrQA, introduced by Chen et al. [7], is a typical pipeline-based multi-passage MRC model. In retrieve component, they utilize TF-IDF to select five relevant Wikipedia articles for each question in SQuAD to narrow the search space. For reader module, they improve the model proposed in 2016 [6] with rich word representations and a pointer module to predict the begin and end position of answer spans. To make scores of candidate spans throughout different passages comparable, Chen et al. utilize unnormalized exponential and argmax function to choose the best answer. In this approach, retrieval and reading are performed separately, but errors made in retrieval stage are easily propagated to the next reading component which leads to performance degradation.

To alleviate error propagation caused by poor document retrieval, one way is to introduce the ranker component, the other is to jointly train retrieval and reading process.

In terms of ranker component, which re-ranks the documents retrieved by search engine, Htut et al. [27] introduce two different ranker, InferSent Ranker and Relation-Networks Ranker. The first one utilizes a feed-forward network to measure the general semantic similarity between the context and question while the relation-networks are applied in the second one to capture the local interactions between context words and question words. Inspired by *Learning to Rank* research, Lee et al. [37] proposed Paragraph Ranker mechanism, which use bi-directional LSTM to compute representations

of passages and questions and measure the similarity between the passages and questions by dot product to score each passage.

For joint training, Reinforced Ranker-Reader (R^3), proposed by Wang et al. [86], is the representative model. In R^3, Match-LSTM [85] is applied to compute the similarity between question and each passage to obtain document representations, which are later fed to both the ranker and reader. In the ranker module, reinforcement learning is utilized to select the most relevant passage, while the function of the reader is to predict the answer span from this selected passage. These two tasks are training jointly to mitigate error propagation caused by wrong document retrieval.

However, retrieval component in above models are in low efficiency. For example, DrQA [7] simply utilizes traditional IR approaches in retrieval component, and R^3 [86] applies question-dependent passage representations to rank the passages. The computational complexity increases with documents corpus becoming larger. In order to accelerate the retrieval process, Das et al. [16] propose a fast and efficient retrieval methods. They represent passages independent from questions and store outputs offline. When given the question, the model computes fast inner product to measure the similarity between passages and question. Then the top ranked passages are fed to the reader to extract answers. Another unique characteristic of their work is iterative interaction between the retriever and reader. They introduce a gated recurrent unit to reformulate query representations taking the state of reader and original query into account. The new query representations are then used to retrieve other relevant passages, which facilitates reread process across corpus.

In multi-passage setting, there may be more than one possible answers among which some are not the right answers to the question. Instead of selecting the first match span as the right answer, Pang et al. [55] propose three other heuristic methods. RAND operation treats all answer spans equally and chooses one randomly from them while MAX operation chooses the one with maximum probability and can be used if there are noisy paragraphs. Moreover, SUM operation assumes that there are more than one spans can be regarded as ground-truth and sums all spans probability together. Similar to MAX operation, Clark & Gardner [13] regard all labeled answer spans as correct at first and inspired by Attention Sum Reader [33], they utilize a summed objective function to choose the one with maximum probability to be the correct answer. In contrast, Lin et al. [40] introduce a fast paragraph selector to filter out passages with wrong answer labels before feeding them to the reader module. They firstly utilize multi-layer perceptron or RNNs to obtain hidden representations of passages and question, respectively. In addition, a self-attention operation is applied to the questions to illustrate their different importance. Then the similarity between the passages and questions is calculated and the top similar ones will be chosen as relevant passages fed to the reader module.

Wang et al. [87] see the significance of evidence aggregation in multi-passage MRC tasks. In their point of view, on the one hand, the correct answers have more evidence appearing across different passages, On the other hand, some questions require various aspects of evidence to answer that. To make full use of multiple evidence, they propose strength-based re-ranker and coverage-based re-ranker. In the first mechanism, the answer with the highest count number of occurrence among the candidates is chosen to be the correct one. The second re-ranker concatenates all passages that contain candidate answers as a new context and feeds that to the reader to obtain the answer which aggregates different aspects of evidence.

To sum up, compared to MRC tasks, multi-passage MRC is much closer to real world application. With several documents given as resources, there are more evidence for answer prediction, thus even the question is complicated, the model can give the answer fairly well. Related documents retrieval is important to multi-passage MRC and evidence aggregated from documents may be complementary or contradict to each other. Hence, free answering in which answers are not limited to the subsequence in the original context is common in multi-passage MRC tasks. Taking advantage of multiple documents and generating answers with right logic and clear semantic to well answer questions still needs a long way to go.

5.4 Conversational Question Answering

MRC requires to answer the question based on the understanding of given passage, whose questions are usually isolated from each other. However, the most natural way that people acquire knowledge is via a series of interrelated question answering process. When given a document, people firstly ask a question and the other one gives an answer. Then based on the answer, another related question

is asked for deeper understanding. This process is performed iteratively which can be regarded as multi-turn conversation. After incorporating conversation into MRC, conversational question answering (CQA) has gradually become the research hotspot.

The definition of CQA can be formulated as below:

CQA

Given the context C, the conversation history with previous questions and answers $H = \{q_1, a_1, \cdots, q_{i-1}, a_{i-1}\}$ and the current question q_i, the CQA task is to predict the right answer a_i by maximizing the conditional probability $P(a_i|C, H, q_i)$.

Many researchers try to create new datasets with given passages and a series of conversation to satisfy the requirement of CQA tasks. Reddy et al. [65] release CoQA, a **Co**nversational **Q**uestion **A**nswering dataset with 8,000 conversations about passages from seven different domains. In the CoQA, a questioner asks questions based on the given passage and an answerer gives answers, which simulates a conversation between two humans when reading the passage. There is no limit to answer form on CoQA which requires more context reasoning. Similarly, Choi et al. [12] introduce QuAC for question answering in context. Compared to the CoQA, passages are only given to the answerer, whereas the questioner asks the questions based on the title of passages. The answerer answers the question with subsequence of original passage and determines whether the questioner can ask a follow up question. Ma et al. [44] extend cloze test tasks to the conversational setting. They utilize the dialogs among characters, selected from transcripts of the TV show *Friends*, to generate related context and ask to fill the blanks with character name according to utterances and context. Different from above two datasets, it aims at multi-party dialog and pays much attention to the doer of some actions. To illustrate CQA task more specific, some examples in the CoQA datasets are presented in Table 5.

Table 5: A few examples of the CoQA dataset

CoQA	
Passage:	Jessica went to sit in her rocking chair. Today was her birthday and she was turning 80.Her granddaughter Annie was coming over in the afternoon and Jessica was very excited to see her. Her daughter Melanie and Malanie's husband Josh were coming as well.
Question 1:	Who had a birthday?
Answer 1 :	Jessica
Question 2:	How old would she be?
Answer 2 :	80
Question 3:	Did she plan to have any visitors?
Answer 3 :	Yes
Question 4:	How many?
Answer 4 :	Three
Question 5:	Who?
Answer 5 :	Annie, Melanie and Josh

CQA brings about some new challenges compared to MRC:

- Conversational History

In MRC tasks, questions and answers are only based on the give passages and questions are independent from previous question answering process. Different from that, conversational history plays an important role in CQA. The follow-up question may be closely related to prior questions and answers. To be more concrete, as shown in Table 5, Question 4 and Question 5 are relevant to Question 3. Moreover, Answer 3 can be a verification for Answer 5. To face this challenge, dialog pairs as conversational history are also fed to the CQA systems as inputs.

- Coreference Resolution

Coreference resolution is a traditional task in natural language processing and it is even more challenging in CQA. Coreference phenomenon may not only occur in the context but appear in the question and answer sentences as well. Coreference can be sorted into two kinds: explicit and implicit. For explicit coreference, there are explicit markers, such as some personal pronouns. For instance, to answer Question 1 *Who had a birthday* in Table 5, the model has to figure out that *"her"* in *Today was her birthday* refers to *Jessica*. Similarly, the understanding of Question 2 is based on the knowledge that *she* means *Jessica*. Compared to explicit coreference, implicit one without explicit markers is much harder to be figured out. Short questions with certain intentions that implicitly refer to previous content is a kind of implicit coreference. For example, to figure out the complete expression of Question 4 (*How many are the visitors?*), the model should extract the correlation between Question 4 and Question 3.

In recent two years, some researchers have made efforts to tackle above new challenges in CQA tasks. Reddy et al. [65] propose a hybrid model, DrQA+PGNet, which combines the sequence-to-sequence model and machine reading comprehension model together to extract and generate answers. To integrate information of conversational history, they treat previous question-answer pairs as sequence and append them to the context. Yatskar et al. [98] utilize an improved MRC models, BiDAF++ with ELMo [59] to answer the question based on the given context and conversational history. Rather than encoding previous dialog information to the context representation, they label answers to previous questions in the context. Instead of simply concatenating previous question-answer pairs as inputs, Huang et al. [30] introduce a flow mechanism to deeply understand conversational history, which encodes hidden context representations during the process of answering previous questions. Similar to Reddy et al. [65], Zhu et al. [103] append previous question-answer pairs to the current question, but in order to find out related conversational history, they employ additional self-attention on questions.

CQA tasks, which incorporate conversation into MRC, is in line with the general process that human understand one thing in the real world. Although researchers have been aware of the significance of conversational information and succeeded in representing conversational history, there is little work on coreference resolution. If the coreference cannot be figured out correctly, it will result in performance degradation. The common coreference phenomena in CQA make this task far more challenging.

6 Open Issues

Based on the aforementioned analyses of literatures, we can observe that there are still some open issues that require future exploration. The most important issue in MRC is that the machine is not genuinely understand the given text as existing MRC models mainly rely on semantic matching to answer the question. As experiments performed by Kaushik & Lipton [34] show that some MRC models perform unexpectedly well when being provided with just passage or question. Although, with the efforts made by researchers in this field, some MRC models outperform human on representative datasets like SQuAD recently, there is still a giant gap between MRC and human comprehension in the following aspects:

- Incorporation of External Knowledge

As an essential component of human intelligence, accumulated common sense or background knowledge is usually utilized in human reading comprehension. To mimic this, knowledge-based MRC is proposed to improve the performance of machine reading with external knowledge. However, how to effectively introduce external knowledge and make full use of it still remains an ongoing challenge. On the one hand, the structure of knowledge stored in knowledge base is so different from the text in the context and question that it is difficult to integrate them together. On the other hand, the performance of knowledge-based MRC is closely related to the quality of knowledge base. The construction of knowledge base is time-consuming which asks for considerable human efforts. In addition, knowledge in knowledge base is sparse, in most of time, relevant external knowledge cannot be found directly to support answer prediction and further reasoning is required. Research on the effective fusion of knowledge graph and machine reading comprehension needs to be further investigated.

- Robustness of MRC Systems

As Jia & Liang [31] point out, most existing MRC models based on word overlap are weak to adversarial question-answer pairs. For SQuAD, they add distracting sentences to the given context, which have semantic overlap with the question and might confuse models but do not contradict to the right answer. With such adversarial examples, performance of MRC systems drops dramatically which reflects that the machine cannot really understand natural language. Although the introduction of answer verification components can alleviate side-effect of plausible answers in a way, the robustness of MRC systems should be enhanced to face the challenge in such adversarial circumstance.

 - Limitation of Given Context

Similar to reading comprehension in language proficiency tests, machine reading comprehension asks the machine to answer questions based on the given context. Such context is a necessity in MRC tasks, but restricts its application. In the real world, the machine is not expected to help students with their reading comprehension exams, but make question answering systems or dialog systems smarter. Efforts made in multi-passage MRC research break the limitation of given context in a way, but there is still a long way to go as how to find out most relevant resources for MRC systems effectively determines the performance of answer prediction. It calls for deeper combination between information retrieval and machine reading comprehension in the future.

 - Lack of Inference Ability

As mentioned before, most existing MRC systems mainly based on the semantic matching between the context and question to give the answer, which results in MRC models being incapable of reasoning. As example given by Liu et al. [41] shows, given the context that *five people on board and two people on the ground died*, the machine cannot infer the right answer *seven* to the question *how many people died* because of the lack of inference ability. How to enable the machine with inference ability still requires further research.

7   Conclusion

This article presents a comprehensive survey on the progresses of neural machine reading comprehension. Based on the thorough analysis of recent work, we give the specific definition of MRC tasks and compare them with each other in depth. The general architecture of neural MRC models is decomposed into four modules and prominent approaches utilized in each module is introduced detailedly. In addition, considering the limitations of MRC, we shed light on some new trends and discuss open issues in this research field.

References

[1] Ricardo Baeza-Yates, Berthier de Araújo Neto Ribeiro, et al. *Modern information retrieval*. New York: ACM Press; Harlow, England: Addison-Wesley,, 2011.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[3] Ondrej Bajgar, Rudolf Kadlec, and Jan Kleindienst. Embracing data abundance: Booktest dataset for reading comprehension. *arXiv preprint arXiv:1610.00956*, 2016.

[4] Akshay Chaturvedi, Onkar Pandit, and Utpal Garain. Cnn for text-based multiple choice question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 272–277, 2018.

[5] Danqi Chen. *Neural Reading Comprehension and Beyond*. PhD thesis, Stanford University, 2018.

[6] Danqi Chen, Jason Bolton, and Christopher D Manning. A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2358–2367, 2016.

[7] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, 2017.

[8] Zheqian Chen, Rongqin Yang, Bin Cao, Zhou Zhao, Deng Cai, and Xiaofei He. Smarnet: Teaching machines to read and comprehend like human. *arXiv preprint arXiv:1710.02772*, 2017.

[9] Zhipeng Chen, Yiming Cui, Wentao Ma, Shijin Wang, and Guoping Hu. Convolutional spatial attention model for reading comprehension with multiple-choice questions. *arXiv preprint arXiv:1811.08610*, 2018.

[10] Hao Cheng, Hao Fang, Xiaodong He, Jianfeng Gao, and Li Deng. Bi-directional attention with agreement for dependency parsing. *arXiv preprint arXiv:1608.02076*, 2016.

[11] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[12] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. Quac: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, 2018.

[13] Christopher Clark and Matt Gardner. Simple and effective multi-paragraph reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 845–855, 2018.

[14] Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 593–602, 2017.

[15] Yiming Cui, Ting Liu, Zhipeng Chen, Shijin Wang, and Guoping Hu. Consensus attention-based neural networks for chinese reading comprehension. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1777–1786, 2016.

[16] Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. Multi-step retriever-reader interaction for scalable open-domain question answering. *arXiv preprint arXiv:1905.05733*, 2019.

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[18] Bhuwan Dhingra, Hanxiao Liu, Ruslan Salakhutdinov, and William W Cohen. A comparative study of word embeddings for reading comprehension. *arXiv preprint arXiv:1703.00993*, 2017.

[19] Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. Gated-attention readers for text comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1832–1846, 2017.

[20] Bhuwan Dhingra, Kathryn Mazaitis, and William W Cohen. Quasar: Datasets for question answering by search and reading. *arXiv preprint arXiv:1707.03904*, 2017.

[21] Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*, 2017.

[22] Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. *arXiv preprint arXiv:1302.4389*, 2013.

[23] Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, et al. Dureader: a chinese machine reading comprehension dataset from real-world applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 37–46, 2018.

[24] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701, 2015.

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

[26] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[27] Phu Mon Htut, Samuel R Bowman, and Kyunghyun Cho. Training a ranking function for open-domain question answering. *NAACL HLT 2018*, page 120, 2018.

[28] Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. Reinforced mnemonic reader for machine reading comprehension. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4099–4106. AAAI Press, 2018.

[29] Minghao Hu, Yuxing Peng, Zhen Huang, Nan Yang, Ming Zhou, et al. Read+ verify: Machine reading comprehension with unanswerable questions. *arXiv preprint arXiv:1808.05759*, 2018.

[30] Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. Flowqa: Grasping flow in history for conversational machine comprehension. *arXiv preprint arXiv:1810.06683*, 2018.

[31] Robin Jia and Percy Liang. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, 2017.

[32] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1601–1611, 2017.

[33] Rudolf Kadlec, Martin Schmid, Ondřej Bajgar, and Jan Kleindienst. Text understanding with the attention sum reader network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 908–918, 2016.

[34] Divyansh Kaushik and Zachary C Lipton. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015, 2018.

[35] Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gáabor Melis, and Edward Grefenstette. The narrativeqa reading comprehension challenge. *Transactions of the Association of Computational Linguistics*, 6:317–328, 2018.

[36] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, 2017.

[37] Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. Ranking paragraphs for improving answer recall in open-domain question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 565–569, 2018.

[38] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. *arXiv preprint arXiv:1706.04115*, 2017.

[39] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004.

[40] Yankai Lin, Haozhe Ji, Zhiyuan Liu, and Maosong Sun. Denoising distantly supervised open-domain question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1736–1745, 2018.

[41] Shanshan Liu, Sheng Zhang, Xin Zhanga, and Hui Wang. R-trans: Rnn transformer network for chinese machine reading comprehension. *IEEE Access*, 2019.

[42] Teng Long, Emmanuel Bengio, Ryan Lowe, Jackie Chi Kit Cheung, and Doina Precup. World knowledge for reading comprehension: Rare entity prediction with hierarchical lstms using external descriptions. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 825–834, 2017.

[43] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[44] Kaixin Ma, Tomasz Jurczyk, and Jinho D Choi. Challenging reading comprehension on daily conversation: Passage completion on multiparty dialog. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2039–2048, 2018.

[45] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pages 6294–6305, 2017.

[46] Todor Mihaylov and Anette Frank. Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 821–832, 2018.

[47] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[48] Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1409, 2016.

[49] Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. Efficient and robust question answering from minimal context over documents. *arXiv preprint arXiv:1805.08092*, 2018.

[50] Preslav Nakov, Alan Ritter, Sara Rosenthal, Fabrizio Sebastiani, and Veselin Stoyanov. Semeval-2016 task 4: Sentiment analysis in twitter. In *Proceedings of the 10th international workshop on semantic evaluation (semeval-2016)*, pages 1–18, 2016.

[51] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*, 2016.

[52] Takeshi Onishi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. Who did what: A large-scale person-centered cloze dataset. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2230–2235, 2016.

[53] Simon Ostermann, Ashutosh Modi, Michael Roth, Stefan Thater, and Manfred Pinkal. Mcscript: A novel dataset for assessing machine comprehension using script knowledge. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, 2018.

[54] Boyuan Pan, Hao Li, Zhou Zhao, Bin Cao, Deng Cai, and Xiaofei He. Memen: Multilayer embedding with memory networks for machine comprehension. *arXiv preprint arXiv:1707.09098*, 2017.

[55] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Lixin Su, and Xueqi Cheng. Has-qa: Hierarchical answer spans model for open-domain question answering. *arXiv preprint arXiv:1901.03866*, 2019.

[56] Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernandez. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1525–1534, 2016.

[57] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[58] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[59] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[60] Boyu Qiu, Xu Chen, Jungang Xu, and Yingfei Sun. A survey on neural machine reading comprehension. *arXiv preprint arXiv:1906.03824*, 2019.

[61] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

[62] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1:8, 2019.

[63] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you dont know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 784–789, 2018.

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

[65] Siva Reddy, Danqi Chen, and Christopher D Manning. Coqa: A conversational question answering challenge. *arXiv preprint arXiv:1808.07042*, 2018.

[66] Matthew Richardson, Christopher JC Burges, and Erin Renshaw. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, 2013.

[67] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533, 1986.

[68] Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*, 2015.

[69] Amrita Saha, Rahul Aralikatte, Mitesh M Khapra, and Karthik Sankaranarayanan. Duorc: Towards complex language understanding with paraphrased reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1683–1693, 2018.

[70] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

[71] Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1047–1055. ACM, 2017.

[72] Alessandro Sordoni, Philip Bachman, Adam Trischler, and Yoshua Bengio. Iterative alternating neural attention for machine reading. *arXiv preprint arXiv:1606.02245*, 2016.

[73] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

[74] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.

[75] Fu Sun, Linyang Li, Xipeng Qiu, and Yang Liu. U-net: Machine reading comprehension with unanswerable questions. *arXiv preprint arXiv:1810.06638*, 2018.

[76] Yibo Sun, Daya Guo, Duyu Tang, Nan Duan, Zhao Yan, Xiaocheng Feng, and Bing Qin. Knowledge based machine reading comprehension. *arXiv preprint arXiv:1809.04267*, 2018.

[77] Simon Suster and Walter Daelemans. Clicr: a dataset of clinical case reports for machine reading comprehension. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1551–1563, 2018.

[78] Chuanqi Tan, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. S-net: From answer extraction to answer generation for machine reading comprehension. *arXiv preprint arXiv:1706.04815*, 2017.

[79] Chuanqi Tan, Furu Wei, Qingyu Zhou, Nan Yang, Weifeng Lv, and Ming Zhou. I know there is no answer: Modeling answer validation for machine reading comprehension. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 85–97. Springer, 2018.

[80] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. Newsqa: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, 2017.

[81] Adam Trischler, Zheng Ye, Xingdi Yuan, Philip Bachman, Alessandro Sordoni, and Kaheer Suleman. Natural language comprehension with the epireader. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 128–137, 2016.

[82] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[83] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700, 2015.

[84] Chao Wang and Hui Jiang. Exploring machine reading comprehension with explicit knowledge. *arXiv preprint arXiv:1809.03449*, 2018.

[85] Shuohang Wang and Jing Jiang. Machine comprehension using match-lstm and answer pointer. *arXiv preprint arXiv:1608.07905*, 2016.

[86] Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. R 3: Reinforced ranker-reader for open-domain question answering. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[87] Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Cambell. Evidence aggregation for answer re-ranking in open-domain question answering.

[88] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198, 2017.

[89] Yequan Wang, Minlie Huang, Li Zhao, et al. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615, 2016.

[90] Zhiguo Wang, Haitao Mi, Wael Hamza, and Radu Florian. Multi-perspective context matching for machine comprehension. *arXiv preprint arXiv:1612.04211*, 2016.

[91] Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*, 2015.

[92] Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014.

[93] Qizhe Xie, Guokun Lai, Zihang Dai, and Eduard Hovy. Large-scale cloze test dataset designed by teachers. *arXiv preprint arXiv:1711.03225*, 2017.

[94] Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016.

[95] Caiming Xiong, Victor Zhong, and Richard Socher. Dcn+: Mixed objective and deep residual coattention for question answering. *arXiv preprint arXiv:1711.00106*, 2017.

[96] Bishan Yang and Tom Mitchell. Leveraging knowledge bases in lstms for improving machine reading. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1436–1446, 2017.

[97] Zhilin Yang, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W Cohen, and Ruslan Salakhutdinov. Words or characters? fine-grained gating for reading comprehension. *arXiv preprint arXiv:1611.01724*, 2016.

[98] Mark Yatskar. A qualitative comparison of coqa, squad 2.0 and quac. *arXiv preprint arXiv:1809.10735*, 2018.

[99] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541*, 2018.

[100] Seunghak Yu, Sathish Reddy Indurthi, Seohyun Back, and Haejun Lee. A multi-stage memory augmented neural network for machine reading comprehension. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 21–30, 2018.

[101] Yang Yu, Wei Zhang, Kazi Hasan, Mo Yu, Bing Xiang, and Bowen Zhou. End-to-end answer chunk extraction and ranking for reading comprehension. *arXiv preprint arXiv:1610.09996*, 2016.

[102] Junbei Zhang, Xiaodan Zhu, Qian Chen, Lirong Dai, Si Wei, and Hui Jiang. Exploring question understanding and adaptation in neural-network-based question answering. *arXiv preprint arXiv:1703.04617*, 2017.

[103] Chenguang Zhu, Michael Zeng, and Xuedong Huang. Sdnet: Contextualized attention-based deep network for conversational question answering. *arXiv preprint arXiv:1812.03593*, 2018.

[104] Haichao Zhu, Furu Wei, Bing Qin, and Ting Liu. Hierarchical attention flow for multiple-choice reading comprehension. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.