Machine Reading Comprehension: The Role of Contextualized Language Models and Beyond

Zhuosheng Zhang
Shanghai Jiao Tong University
Department of Computer Science and Engineering
zhangzs@sjtu.edu.cn

Hai Zhao
Shanghai Jiao Tong University
Department of Computer Science and Engineering
zhaohai@cs.sjtu.edu.cn

Rui Wang
National Institute of Information and Communications Technology (NICT)
wangrui@nict.go.jp

Machine reading comprehension (MRC) aims to teach machines to read and comprehend human languages, which is a long-standing goal of natural language processing (NLP). With the burst of deep neural networks and the evolution of contextualized language models (CLMs), the research of MRC has experienced two significant breakthroughs. MRC and CLM, as a phenomenon, have a great impact on the NLP community. In this survey, we provide a comprehensive and comparative review on MRC covering overall research topics about 1) the origin and development of MRC and CLM, with particular focus on the role of CLMs; 2) the impact of MRC and CLM to the NLP community; 3) the definition, datasets, and evaluation of MRC; 4) general MRC architecture and technical methods in the view of two-stage Encoder-Decoder solving architecture from the insights of the cognitive process of humans; 5) previous highlights, emerging topics, and our empirical analysis, among which we especially focus on what works in different periods of MRC researches. We propose a full-view categorization and new taxonomies on these topics. The primary views we have arrived at are that 1) MRC boosts the progress from language processing to understanding; 2) the rapid improvement of MRC systems greatly benefits from the development of CLMs; 3) the theme of MRC is gradually moving from shallow text matching to cognitive reasoning.

1. Introduction

Natural language processing (NLP) tasks can be roughly divided into two categories: 1) fundamental NLP, including language modeling and representation, and linguistic structure and analysis, including morphological analysis, word segmentation, syntactic, semantic and discourse paring, etc.; 2) application NLP, including machine question answering, dialogue system, machine translation, and other language understanding and inference tasks. With the rapid development of NLP, natural language understanding (NLU) has aroused broad interests, and a series of NLU tasks have emerged. In the early days, NLU was regarded as the next stage of NLP. With more computation resources available, more complex networks become possible, and researchers

are inspired to move forward to the frontier of human-level language understanding. Inevitably, machine reading comprehension (MRC) (Richardson, Burges, and Renshaw 2013; Hermann et al. 2015; Hill et al. 2015; Rajpurkar et al. 2016) as a new typical task has boomed in the field of NLU. Figure 1 overviews MRC in the background of language processing and understanding.

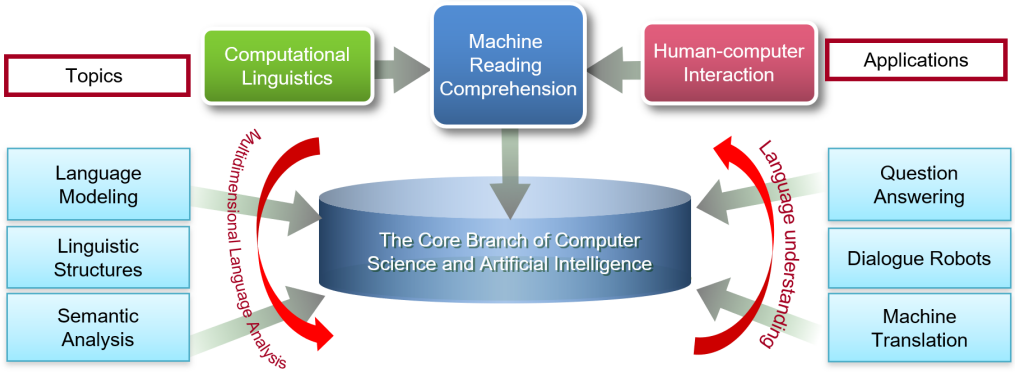

Figure 1: Overview of language processing and understanding.

MRC is a long-standing goal of NLU that aims to teach a machine to read and comprehend textual data. It has significant application scenarios such as question answering and dialog systems (Choi et al. 2018; Reddy, Chen, and Manning 2019; Zhang et al. 2018c; Zhu et al. 2018b; Xu et al. 2020). The related MRC research can be traced back to the studies of story comprehension (Lehnert 1977; Cullingford 1977). After decades of decline, MRC becomes a hot research topic recently and experiences rapid development. MRC has a critical impact on NLU and the broader NLP community. As one of the major and challenging problems of NLP concerned with comprehensive knowledge representation, semantic analysis, and reasoning, MRC stimulates great research interests in the last decade. The study of MRC has experienced two significant peaks, namely, 1) the burst of deep neural networks; 2) the evolution of contextualized language models (CLMs). Figure 2 shows the research trend statistics of MRC and CLMs in the past five years.

Early MRC task was simplified as requiring systems to return a sentence that contains the right answer. The systems are based on rule-based heuristic methods, such as bag-of-words approaches (Hirschman et al. 1999), and manually generated rules (Riloff and Thelen 2000; Charniak et al. 2000). With the introduction of deep neural networks and effective architecture like attention mechanisms in NLP (Bahdanau, Cho, and Bengio 2014; Hermann et al. 2015), the research interests of MRC boomed since around 2015 (Chen, Bolton, and Manning 2016; Bajaj et al. 2016; Rajpurkar et al. 2016; Trischler et al. 2017; Dunn et al. 2017; He et al. 2018; Kočiskỳ et al. 2018; Yang et al. 2018; Reddy, Chen, and Manning 2019; Pan et al. 2019a). The main topics were fine-grained text encoding and better passage and question interactions (Seo et al. 2017; Yang et al. 2017a; Dhingra et al. 2017; Cui et al. 2017; Zhang et al. 2018b).

CLMs lead to a new paradise of contextualized language representations — using the whole sentence-level representation for language modeling as pre-training, and the context-dependent hidden states from the LM are used for downstream task-specific fine-tuning. Deep pre-trained CLMs (Peters et al. 2018; Devlin et al. 2018; Yang et al. 2019c; Lan et al. 2019; Dong et al. 2019; Clark et al. 2019c; Joshi et al. 2020) greatly

strengthened the capacity of language encoder, the benchmark results of MRC were boosted remarkably, which stimulated the progress towards more complex reading, comprehension, and reasoning systems (Welbl, Stenetorp, and Riedel 2018; Yang et al. 2018; Ding et al. 2019). As a result, the researches of MRC become closer to human cognition and real-world applications. On the other hand, more and more researchers are interested in analyzing and interpreting how the MRC models work, and investigating the *real* ability beyond the datasets, such as performance in the adversarial attack (Jia and Liang 2017; Wallace et al. 2019), as well as the benchmark capacity of MRC datasets (Sugawara et al. 2018, 2019; Schlegel et al. 2020). The common concern is the overestimated ability of MRC systems, which shows to be still in a shallow comprehension stage drawn from superficial pattern-matching heuristics. Such assessments of models and datasets would be suggestive for next-stage studies of MRC methodologies.

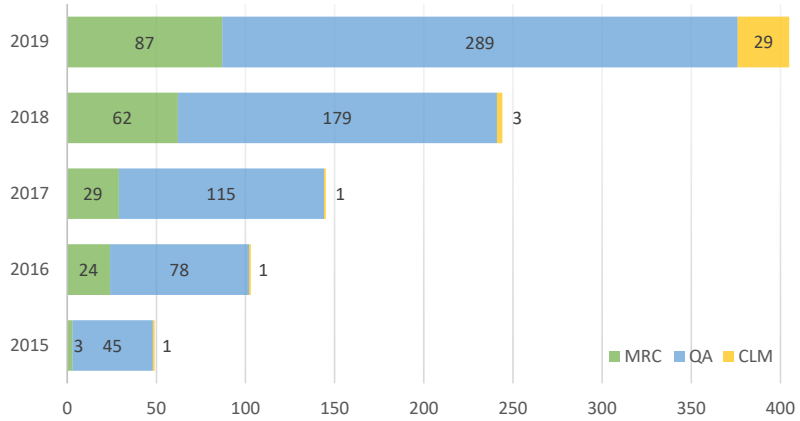

Figure 2: The number of papers concerning MRC, QA, and CLM collected from 2015 to 2019. The search terms are MRC: {machine reading comprehension, machine comprehension, machine comprehend, mrc}; QA: {question answering, qa}. Since MRC papers are often in the name of QA, we also present the QA papers for reference. MRC and QA papers are searched by keywords in paper titles on `https://arxiv.org`. CLM statistics are calculated based on the influential open-source repository: `https://github.com/thunlp/PLMpapers`.

MRC is a generic concept to probe for language understanding capabilities (Schlegel et al. 2020; Gardner et al. 2019). In the early stage, MRC was regarded as the form of triple-style (passage, question, answer) question answering (QA) task, such as the cloze-style (Hermann et al. 2015; Hill et al. 2015), multiple-choice (Lai et al. 2017; Sun et al. 2019a), and span-QA (Rajpurkar et al. 2016; Rajpurkar, Jia, and Liang 2018). In recent years, we witness that the concept of MRC has evolved to a broader scope, which caters to the theme of language understanding based interaction and reasoning, in the form of question answering, text generation, conversations, etc. Though MRC originally served as the form of question answering, it can be regarded as not only just the extension of QA but also a new concept used for studying the capacity of language understanding over some context that is close to cognitive science, instead of a single task itself. Regarding MRC as phenomenon, there is a new emerging interest showing that classic NLP tasks can be cast as span-QA MRC form, with modest performance gains than previous methodologies (McCann et al. 2018; Keskar et al. 2019; Li et al. 2019b,a; Keskar et al. 2019; Gao et al. 2019a, 2020).

Although it is clear that computation power substantially fuels the capacity of MRC systems in the long run, building simple, explainable, and practical models is equally essential for real-world applications. It is instructive to review the prominent highlights in the past. The generic nature, especially what works in the past and the inspirations of MRC to the NLP community, would be suggestive for future studies, which are the focus of discussions in this work.

This work reviews MRC covering the scope of background, definition, influence, datasets, technical and benchmark success, empirical assessments, current trends, and future opportunities. Our main contributions are summarized as follows:

- **Comprehensive review and in-depth discussions**. We conduct a comprehensive review of the origin and the development of MRC, with a special focus on the role of CLMs. We propose new taxonomies of the technical architecture of MRC, by formulating the MRC systems as two-stage solving architecture in the view of cognition psychology and provide a comprehensive discussion of research topics to gain insights. By investigating typical models and the trends of the main flagship datasets and leaderboards concerning different types of MRC, along with our empirical analysis, we provide observations of the advances of techniques in different stages of studies.
- **Wide coverage on highlights and emerging topics**. MRC has experienced rapid development. We present a wide coverage of previous highlights and emerging topics, including casting traditional NLP tasks into MRC formation, multiple granularity feature modeling, structured knowledge injection, contextualized sentence representation, matching interaction, and data augmentation.
- **Outlook on the future**. This work summaries the trends and discussions for future researches, including interpretability of datasets and models, decomposition of prerequisite skills, complex reasoning, large-scale comprehension, low-resource MRC, multimodal semantic grounding, and deeper but efficient model design.

The remainder of this survey is organized as follows: first, we present the background, categorization, and derivatives of CLM and discuss the mutual influence between CLM and MRC in §2; an overview of MRC including the impact to general NLP scope, formations, datasets, and evaluation metrics is given in §3; then, we discuss the technical methods in the view of two-stage solving architecture, and summarize the major topics and challenges in §4; next, our work goes deeper in §5 to discover what works in different stages of MRC, by reviewing the trends and highlights entailed in the typical MRC models. Our empirical analysis is also reported for the verification of simple and effective tactic optimizations based on the strong CLMs; finally, we discuss the trends and future opportunities in §6, together with conclusions in §7;

2. The Role of Contextualized Language Model

2.1 From Language Model to Language Representation

Language modeling is the foundation of deep learning methods for natural language processing. Learning word representations has been an active research area, and aroused great research interests for decades, including non-neural (Brown et al. 1992; Ando and Zhang 2005; Blitzer, McDonald, and Pereira 2006) and neural methods (Mikolov et al. 2013; Pennington, Socher, and Manning 2014). Regarding language modeling, the basic topic is n-gram language model (LM). An n-gram Language model is a probability distribution over word (n-gram) sequences, which can be regarded

Table 1: Comparison of language representation.

Model	Repr. form	Context	Training object	Usage
n-gram LM	One-hot	Sliding widow	n-gram LM (MLE)	Lookup
Word2vec/GloVe	Embedding	Sliding widow	n-gram LM (MLE)	Lookup
Contextualized LM	Embedding	Sentence	n-gram LM (MLE), +ext	Fine-tune

with a training objective of predicting unigram from $(n-1)$-gram. Neural networks use continuous and dense representation, or further embedding of words to make their predictions, which is effective for alleviating the curse of dimensionality – as language models are trained on larger and larger texts, the number of unique words increases.

Compared with the word embeddings learned by Word2Vec (Mikolov et al. 2013) or GloVe (Pennington, Socher, and Manning 2014), sentence is the least unit that delivers complete meaning as human uses language. Deep learning for NLP quickly found it is a frequent requirement on using a network component encoding a sentence input so that we have the *Encoder* for encoding the complete sentence-level context. The encoder can be the traditional RNN, CNN, or the latest Transformer-based architectures, such as ELMo (Peters et al. 2018), GPT$_{v1}$ (Radford et al. 2018), BERT (Devlin et al. 2018), XLNet (Yang et al. 2019c), RoBERTa (Liu et al. 2019c), ALBERT (Lan et al. 2019), and ELECTRA (Clark et al. 2019c), for capturing the contextualized sentence-level language representations.[1] These encoders differ from sliding window input (e.g., that used in Word2Vec) that they cover a full sentence instead of any fixed length sentence segment used by the sliding window. Such difference especially matters when we have to handle passages in MRC tasks, where the passage always consists of a lot of sentences. When the model faces passages, the sentence, instead of word, is the basic unit of a passage. In other words, MRC, as well as other application tasks of NLP, needs a sentence-level encoder, to represent sentences into embeddings, so as to capture the deep and contextualized sentence-level information.

An encoder model can be trained in a style of n-gram language model so that there comes the language representation, which includes four elements: 1) representation form; 2) context; 3) training object (e.g., n-gram language model); 4) usage. For contextualized language representation, the representation for each word depends on the entire context in which it is used, which is dynamic embedding. Table 1 presents a comparison of the three main language representation approaches.

2.2 CLM as Phenomenon

2.2.1 Revisiting the Definition. First, we would like to revisit the definitions of the recent contextualized encoders. For the representative models, ELMo is called *Deep contextualized word representations*, and BERT *Pre-training of deep bidirectional transformers for language understanding*. With the follow-up research goes on, there are studies that call those models as pre-trained (language) models (Sanh et al. 2019; Goldberg 2019). We argue that such a definition is reasonable but not accurate enough. The focus of these models are supposed to be *contextualized* (as that show in the name of ELMo), in

1 This is a non-exhaustive list of important CLMs introduced recently. In this work, our discussions are mainly based on these typical CLMs, which are highly related to MRC researches, and most of the other models can be regarded as derivatives.

terms of the evolution of language representation architectures, and the actual usages of these models nowadays. As a consensus of limited computing resources, the common practice is to fine-tune the model using task-specific data after the public pre-trained sources, so that pre-training is neither the necessary nor the core element. As shown in Table 1, the training objectives are derived from n-gram language models. Therefore, we argue that pre-training and fine-tuning are just the manners we use the models. The essence is the deep contextualized representation from language models; thus, we call these pre-trained models **contextualized language models, CLMs)** in this paper.

2.2.2 Evolution of CLM Training Objectives. In this part, we abstract the inherent relationship of n-gram language model and the subsequent contextualized LM techniques. Then, we elaborate the evolution of the typical CLMs considering the salient role of the training objectives.

Regarding the training of language models, the standard and common practice is using the n-gram language modeling. It is also the core training objective in CLMs. An n-gram Language model yields a probability distribution over text (n-gram) sequences, which is a classic maximum likelihood estimation (MLE) problem. The language modeling is also known as **autoregressive** (AR) scheme.

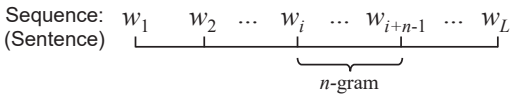

Figure 3: Example of n-grams.

Specifically, given a sequence of n items $\mathbf{w} = w_{i:i+n-1}$ from a text (Figure 3), the probability of the sequence is measured as

$$p(\mathbf{w}) = p(w_i \mid w_{i:i+n-2}), \tag{1}$$

where $p(w_i|w_{i:i+n-2})$ denotes the conditional probability of $p(w_i)$ in the sequence, which can be estimated by the context representation over $w_{i:i+n-2}$. The LM training is performed by maximizing the likelihood:

$$\max_{\theta} \sum_{\mathbf{w}} \log p_{\theta}(\mathbf{w}), \tag{2}$$

where θ denotes the model parameter.

In practice, n-gram models have been shown to be extremely effective in modeling language data, which is a core component in modern language applications. The early contextualized representation is obtained by static word embedding and a network encoder. For example, CBOW and Skip-gram (Mikolov et al. 2013) either predicts the word using context or predict context by word, where the n-gram context is provided by a fixed sliding window. The trained model parameters are output as a word embedding matrix (also known as a lookup table), which contains the context-independent representations for each word in a vocabulary. The vectors are then used in a low-level layer (i.e., embedding layer) of neural network, and an encoder, such as RNN is further used to obtain the contextualized representation for an input sentence.

For recent LM-derived **contextualized** presentations (Peters et al. 2018; Devlin et al. 2018; Yang et al. 2019c), the central point of the subsequent optimizations are concerning

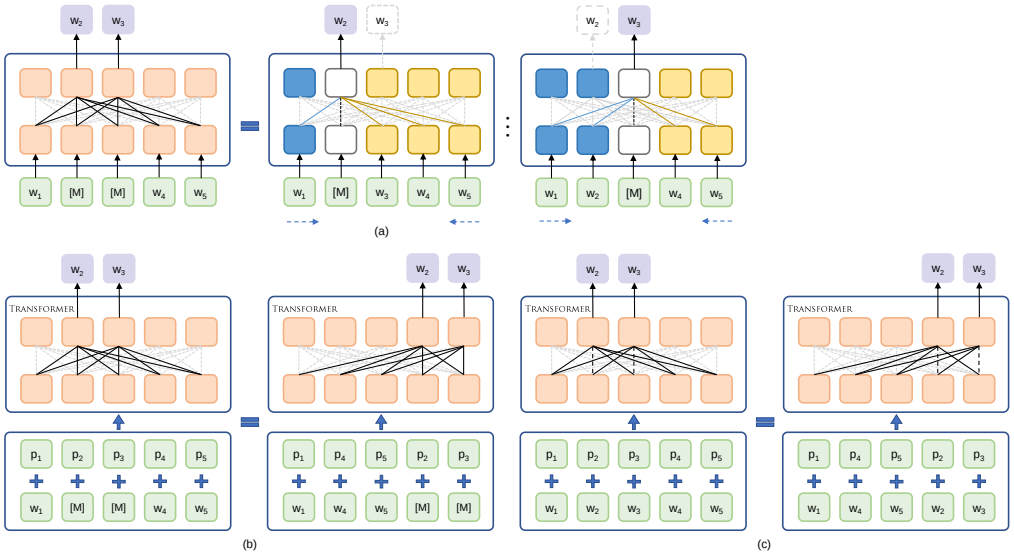

Figure 4: The possible transformation of MLM and PLM, where w_i and p_i represent token and position embeddings. $[M]$ is the special mask token used in MLM. The left side of MLM (a) can be seen as bidirectional AR streams (in blue and yellow, respectively) at the right side. For MLM (b) and PLM (c), the left sides are in original order, and the right sides are in permuted order, which are regarded as a unified view.

the context. They are trained with much larger n-grams that cover a full sentence where n is extended to the sentence length — **when n expands to the maximum, the conditional context thus corresponds to the whole sequence**. The word representations are the function of the entire sentence, instead of the static vectors over a pre-defined lookup table. The corresponding functional model is regarded as a contextualized language model. Such a contextualized model can be directly used to produce context-sensitive sentence-level representations for task-specific fine-tuning. Table 2 shows the comparisons of CLMs.

For an input sentence $\mathbf{s} = w_{1:L}$, we extend the objective of n-gram LM in the context of length L from Equation (2):

$$\sum_{k=c+1}^{L} \log p_\theta(w_k \mid w_{1:k-1}), \tag{3}$$

where c is the cutting point that separate the sequence into a non-target conditional subsequence $k \leq c$ and a target subsequence $k > c$. It can be further written in a bidirectional form:

$$\sum_{k=c+1}^{L} (\log p_\theta(w_k \mid w_{1:k-1}) + \log p_\theta(w_k \mid w_{k+1:L})), \tag{4}$$

Table 2: Comparison of CLMs. NSP: next sentence prediction (Devlin et al. 2018). SOP: sentence order prediction (Lan et al. 2019). RTD: replaced token detection (Clark et al. 2019c).

Model	Loss	2^{nd} Loss	Direction	Encoder arch.	Input
ELMo	n-gram LM	-	Bi	RNN	Char
GPT_{v1}	n-gram LM	-	Uni	Transformer	Subword
BERT	Masked LM	NSP	Bi	Transformer	Subword
RoBERTa	Masked LM	-	Bi	Transformer	Subword
ALBERT	Masked LM	SOP	Bi	Transformer	Subword
XLNet	Permu. n-gram LM	-	Bi	Transformer-XL	Subword
ELECTRA	Masked LM	RTD	Bi	GAN	Subword

which corresponds to the bidirectional LM used in ELMo (Peters et al. 2018). The bidirectional modeling of ELMo is achieved by the concatenation of independently trained forward and backward LSTMs.

To allow simultaneous bidirectional (or non-directional) training, BERT (Devlin et al. 2018) adopted Transformer to process the whole input at once, and proposed Masked LM (MLM) to take advantage of both the left and right contexts. Some tokens in a sentence are randomly replaced with a special mask symbol with a small probability. Then, the model is trained to predict the masked token based on the context. MLM can be seen as a variant of n-gram LM (Figure 4(a)) to a certain extent — bidirectional autoregressive n-gram LM.[2] Let $\mathcal{D}$ denote the set of masked positions using the mask symbol $[M]$. We have $w_{\mathcal{D}}$ as the set of masked tokens, and $\mathbf{s}'$ as the masked sentence. As the example shown in the left part of Figure 4(b), $\mathcal{D} = \{2, 3\}$, $w_{\mathcal{D}} = \{w_2, w_3\}$ and $\mathbf{s}' = \{w_1, [M], w_4, [M], w_5\}$. The objective of MLM is to maximize the following objective:

$$\sum_{k \in \mathcal{D}} \log p_\theta(w_k \mid \mathbf{s}') \tag{5}$$

Compared with Equation (4), it is easy to find that the prediction is based on the whole context in Equation (5) instead of only one direction for each estimation, which indicates the major difference of BERT and ELMo. However, the essential problem in BERT is that the mask symbols are never seen at fine-tuning, which faces a mismatch between pre-training and fine-tuning.

To alleviate the issue, XLNet (Yang et al. 2019c) utilized permutation LM (PLM) to maximize the expected log-likelihood of all possible permutations of the factorization order, which is the AR LM objective.[3] For the input sentence $\mathbf{s} = w_{1:L}$, we have $\mathcal{Z}_L$ as the permutations of set $\{1, 2, \cdots, L\}$. For a permutation $z \in \mathcal{Z}_L$, we split z into a non-target conditional subsequence $z \leq c$ and a target subsequence $z > c$, where c is the cutting point. The objective is to maximize the log-likelihood of the target tokens conditioned

2 In a general view, the idea of MLM can also be derived from CBOW, which is to predict word according to the conditional n-gram surrounding context.

3 In contrast, the language modeling method in BERT is called denoising **autoencoding** (Yang et al. 2019c) (AE). AE can be seen as the natural combination of AR loss and a certain neural network.

on the non-target tokens:

$$\mathbb{E}_{z \in \mathcal{Z}_L} \sum_{k=c+1}^{L} \log p_\theta(w_{z_k} \mid w_{z_{1:k-1}}). \tag{6}$$

The key of both MLM and PLM is predicting word(s) according to a certain context derived from n-grams, which can be modeled in a unified view (Song et al. 2020). In detail, under the hypothesis of word order insensitivity, MLM can be directly unified as PLM when the input sentence is permutable (with insensitive word orders), as shown in Figure 4(b-c). It can be satisfied thanks to the nature of the Transformer-based models, such as BERT and XLNet. Transformer takes tokens and their positions in a sentence as inputs, and it is not sensitive to the absolute input order of these tokens. Therefore, the objective of MLM can be also written as the permutation form,

$$\mathbb{E}_{z \in \mathcal{Z}_L} \sum_{k=c+1}^{L} \log p_\theta(w_{z_k} \mid w_{z_{1:c}}, M_{z_{k:L}}), \tag{7}$$

where $M_{z_{k:L}}$ denote the special mask tokens $[M]$ in positions $z_{k:L}$.

From Equations (3), (6), and (7), we see that MLM and PLM share similar formulations with the n-gram LM with slight difference in the conditional context part in $p(\mathbf{s})$: MLM conditions on $w_{z_{1:c}}$ and $M_{k:L}$, and PLM conditions on $w_{z_{1:k-1}}$. **Both MLM and PLM can be explained by the n-gram LM, and even unified into a general formation**. With similar inspiration, MPNet (Song et al. 2020) combined the Masked LM and Premuted LM for taking both of the advantages.

2.2.3 Architectures of CLMs. So far, there are mainly three leading architectures for language modeling,[4] RNN, Transformer, and Transformer-XL. Figure 5 depicts the three encoder architectures.

RNN. RNN and its derivatives are popular approaches for language encoding and modeling. The widely-used variants are GRU (Cho et al. 2014) and LSTM (Hochreiter and Schmidhuber 1997). RNN models process the input tokens (commonly words or characters) one by one to capture the contextual representations between them. However, the processing speed of RNNs is slow, and the ability to learn long-term dependencies is still limited due to vanishing gradients.

Transformer. To alleviate the above issues of RNNs, Transformer was proposed, which employs *multi-head self-attention* (Vaswani et al. 2017) modules receive a segment of tokens (i.e., subwords) and the corresponding position embedding as input to learn the direct connections of the sequence at once, instead of processing tokens one by one.

Transformer-XL. Though both RNN and Transformer architectures have reached impressive achievements, their main limitation is capturing long-range dependencies. Transformer-XL (Dai et al. 2019) combines the advantages of RNN and Transformer,

4 Actually, CNN also turns out well-performed feature extractor for some NLP tasks like text classification, but RNN is more widely used for MRC, even most NLP tasks; thus we omit the description of CNNs and focus on RNNs as the example for traditional encoders.

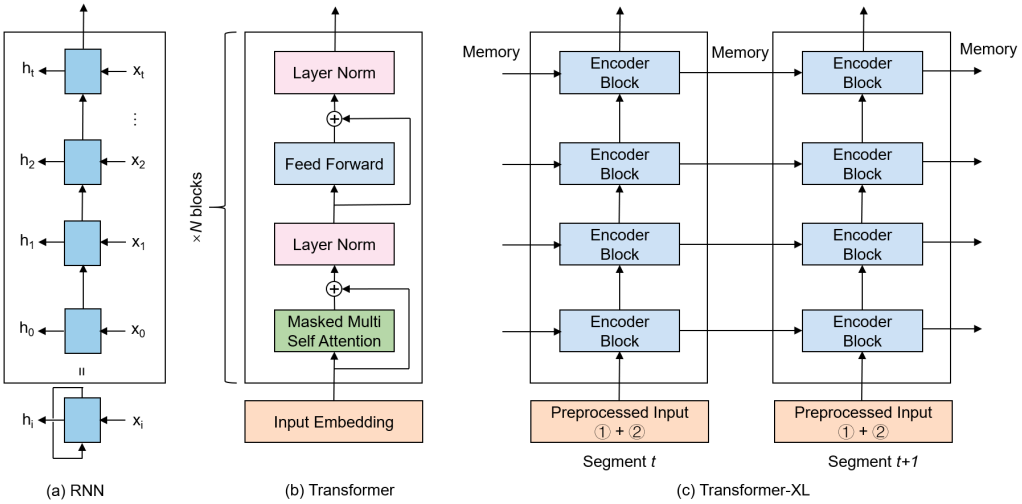

Figure 5: RNN, Transformer, and Transformer-XL encoder architectures for CLMs.

which uses the self-attention modules on each segment of input data and a recurrent mechanism to learn dependencies between consecutive segments. In detail, two new techniques are proposed:

1. **Segment-level Recurrence**. The recurrence mechanism is proposed to model long-term dependencies by using information from previous segments. During training, the representations computed for the previous segment are fixed and cached to be reused as an extended context when the model processes the next new segment. This recurrence mechanism is also effective in resolving the context fragmentation issue, providing necessary context for tokens in the front of a new segment.
2. **Relative Positional Encoding**. The original positional encoding deals with each segment separately. As a result, the tokens from different segments have the same positional encoding. The new relative positional encoding is designed as part of each attention module, as opposed to the encoding position only before the first layer. It is based on the relative distance between tokens, instead of their absolute position.

2.2.4 Derivative of CLMs. Pre-training and fine-tuning have become a new paradigm of NLP, and the major theme is to build a strong encoder. Based on the inspirations of impressive models like ELMo and BERT, a wide range of CLMs derivatives have been proposed. In this part, we discuss various major variants concerning MRC tasks. Table 3 shows the performance comparison of the CLM derivatives. The advances behind these models are in four main topics:

Masking Strategy. The original masking of BERT is based on subword, which would be insufficient for capturing global information using the local subword signals. SpanBERT (Joshi et al. 2020) proposed a random span masking strategy based on geometric distribution, indicating that the proposed masking sometimes works even better than masking linguistically-coherent spans. To avoid using the same mask for each training

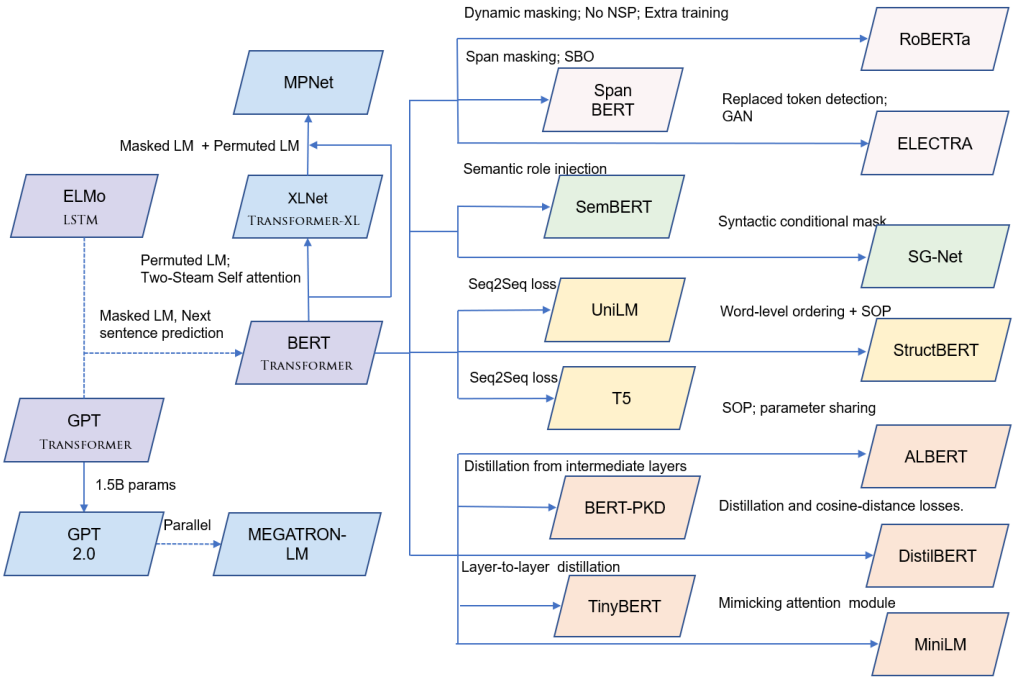

Figure 6: Derivative of CLMs. The main features are noted above the arrow. Solid and dotted arrows indicate the direct and implicit inheritance.

instance in every epoch, RoBERTa (Liu et al. 2019c) used dynamic masking to generate the masking pattern every time feeding a sequence to the model, indicating that dynamic masking would be crucial for pre-training a great many steps or with large-scale datasets. ELECTRA (Clark et al. 2019c) improved the efficiency of masking by adopting a replaced token detection objective.

Knowledge Injection. Extra knowledge can be easily incorporated into CLMs by both embedding fusion and masking. SemBERT (Zhang, Zhao, and Zhou 2020) indicated that fusing semantic role label embedding and word embedding can yield better semantic-level language representation, showing that salient word-level high-level tag features can be well integrated with subword-level token representations. SG-Net (Zhang et al. 2020c) presented a dependency-of-interest masking strategy to use syntax information as a constraint for better linguistics inspired representation.

Training Objective. Besides the core MLE losses that used in language models, some extra objectives were investigated for better adapting target tasks. BERT (Devlin et al. 2018) adopted the next sentence prediction (NSP) loss, which matches the paired form in NLI tasks. To better model inter-sentence coherence, ALBERT (Lan et al. 2019) replaced NSP loss with a sentence order prediction (SOP) loss. StuctBERT (Wang et al. 2020a) further leveraged word-level ordering and sentence-level ordering as structural objectives in pre-training. SpanBERT (Joshi et al. 2020) used span boundary objective (SBO), which requires the model to predict masked spans based on span boundaries, to integrate structure information into pre-training. UniLM (Dong et al. 2019) extended

Table 3: Performance of CLM derivatives. F1 scores for SQuAD1.1 and SQuAD2.0, accuracy for RACE. * indicates results that depend on additional data augmentation. † indicate the result is from Yang et al. (2019c) as it was not reported in the original paper (Devlin et al. 2018). The BERT_{base} result for SQuAD2.0 is from Wang et al. (2020b). The *italic* numers are baselines for calculating the D-values ↑.

Method	SQuAD1.1				SQuAD2.0				RACE	
	Dev	↑ Dev	Test	↑ Test	Dev	↑ Dev	Test	↑ Test	Acc	↑ Acc
ELMo	*85.6*	-	*85.8*	-	-		-		-	-
GPT_{v1}	-	-	-	-	-	-	-	-	*59.0*	-
BERT_{base}	88.5	2.9	-	-	*76.8*		-		65.3	6.3
BERT-PKD	85.3	-0.3	-	-	69.8	-7.0	-	-	60.3	1.3
DistilBERT	86.2	0.6	-	-	69.5	-7.3	-	-		
TinyBERT	87.5	1.9	-	-	73.4	-3.4	-	-	-	-
MiniLM	-	-	-	-	76.4	-0.4	-	-	-	-
Q-BERT	88.4	2.8	-	-	-	-	-	-	-	-
BERT_{large}	91.1*	5.5	91.8*	6	81.9	5.1	83.0	-	72.0†	-
SemBERT_{large}	-	-	-	-	83.6	6.8	85.2	2.2	-	-
SG-Net	-	-	-	-	88.3	11.5	87.9	4.9	74.2	15.2
SpanBERT_{large}	-	-	94.6	8.8	-	-	88.7	5.7	-	-
$\text{StructBERT}_{large}$	92.0	6.4	-	-	-	-	-	-	-	-
RoBERTa_{large}	94.6	9.0	-	-	89.4	12.6	89.8	6.8	83.2	24.2
$\text{ALBERT}_{xxlarge}$	94.8	9.2	-	-	90.2	13.4	90.9	7.9	86.5	27.5
XLNet_{large}	94.5	8.9	95.1*	9.3	88.8	12	89.1*	6.1	81.8	22.8
UniLM	-	-	-	-	83.4	6.6	-	-	-	-
ELECTRA_{large}	94.9	9.3	-	-	90.6	13.8	91.4	8.4	-	-
$\text{Megatron-LM}_{3.9B}$	95.5	9.9	-	-	91.2	14.4	-	-	89.5	30.5
T5_{11B}	95.6	10.0	-	-	-	-	-	-	-	-

the mask prediction task with three types of language modeling tasks: unidirectional, bidirectional, and sequence-to-sequence (Seq2Seq) prediction. The Seq2Seq MLM was also adopted as the objective in T5 (Raffel et al. 2019), which employed a unified Text-to-Text Transformer for general-purpose language modeling. ELECTRA Clark et al. (2019c) proposed new pre-training task — replaced token detection (RTD) and a generator-discriminator model was designed accordingly. The generator is trained to perform MLM, and then the discriminator predicts whether each token in the corrupted input was replaced by a generator sample or not.

Model Optimization. RoBERTa (Liu et al. 2019c) found that the model performance can be substantially improved by 1) training the model longer, with bigger batches over more data can; 2) removing the next sentence prediction objective; 3) training on longer sequences; 4) dynamic masking on the training data. Megatron (Shoeybi et al. 2019) presented an intra-layer model-parallelism approach that can support efficiently training very large Transformer models.

To obtain light-weight yet powerful models for real-world use, model compression is an effective solution. ALBERT (Lan et al. 2019) used cross-layer parameter sharing and factorized embedding parameterization to reduce the model parameters. Knowledge distillation (KD) also aroused hot interests. BERT-PKD proposed a patient KD

Table 4: The initial applications of CLMs. The concerned NLU task can also be regarded as a special case of MRC as discussed in §3.

	NLU		MRC		
	SNLI	GLUE	SQuAD1.1	SQuAD2.0	RACE
ELMo	✓	✗	✓	✗	✗
GPT_{v1}	✓	✓	✗	✗	✓
BERT	✗	✓	✓	✓	✗
RoBERTa	✗	✓	✓	✓	✓
ALBERT	✗	✓	✓	✓	✓
XLNet	✗	✓	✓	✓	✓
ELECTRA	✗	✓	✓	✓	✗

mechanism that learns from multiple intermediate layers of the teacher model for incremental knowledge extraction. DistilBERT (Sanh et al. 2019) leveraged a knowledge distillation mechanism during the pre-training phase, which introduced a triple loss combining language modeling, distillation, and cosine-distance losses. TinyBERT (Jiao et al. 2019) adopted layer-to-layer distillation with embedding outputs, hidden states, and self-attention distributions. MiniLM (Wang et al. 2020b) performed the distillation on self-attention distributions and value relation of the teacherâĂŹs last Transformer layer to guide student model training. Moreover, quantization is another optimization technique by compressing parameter precision. Q-BERT (Shen et al. 2019) applied a Hessian based mix-precision method to compress the mode with minimum loss in accuracy and more efficient inference.

2.3 Correlations Between MRC and CLM

In the view of practice, MRC and CLM are complementary to each other. MRC is a challenging problem concerned with comprehensive knowledge representation, semantic analysis, and reasoning, which arouses great research interests and stimulates the development of wide ranges of advanced models, including CLMs. As shown in Table 4, MRC also serves as an appropriate testbed for language representation, which is the focus of CLMs. On the other hand, the progress of CLM greatly promotes MRC tasks, achieving impressive gains of model performance. With such an indispensable association, human-parity performance has been first achieved and frequently reported after the release of CLMs.

3. MRC as Phenomenon

3.1 Classic NLP Meets MRC

MRC has great inspirations to the NLP tasks. Most NLP tasks can benefit from the new task formation as MRC. The advantage may lie within both sides of 1) strong capacity of MRC-style models, e.g., keeping the pair-wise training mode like the pre-training of CLMs and better-contextualized modeling like multi-turn question answering (Li et al. 2019b); 2) unifying different tasks as MRC formation, and taking advantage of multi-tasking to share and transfer knowledge.

Traditional NLP tasks can be cast as QA-formed reading comprehension over a context, including question answering, machine translation, summarization, natural language inference, sentiment analysis, semantic role labeling, zero-shot relation extraction, goal-oriented dialogue, semantic parsing, and commonsense pronoun resolution (McCann et al. 2018). The span extraction task formation of MRC also leads to superior or comparable performance for standard text classification and regression tasks, including those in GLUE benchmarks (Keskar et al. 2019), and entity and relation extraction tasks (Li et al. 2019b,a; Keskar et al. 2019). As MRC aims to evaluate how well machine models can understand human language, the goal is actually similar to the task of Dialogue State Tracking (DST). There are recent studies that formulate the DST task into MRC form by specially designing a question for each slot in the dialogue state, and propose MRC models for dialogue state tracking (Gao et al. 2019a, 2020).

3.2 MRC Goes Beyond QA

In most NLP/CL papers, MRC is usually organized as a question answering task with respect to a given reference text (e.g., a passage). As discussed in Chen (2018), there is a close relationship between MRC and QA. (Shallow) reading comprehension can be regarded as an instance of question answering, but they emphasize different final targets. We believe that the general MRC is a concept to probe for language understanding capabilities, which is very close to the definition of NLU. In contrast, QA is a format (Gardner et al. 2019), which is supposed to be the actual way to check how the machine comprehends the text. The rationale is the difficulty to measure the primary objective of MRC — evaluating the degree of machine comprehension of human languages. To this end, QA is a fairly simple and effective format. MRC also goes beyond the traditional QA, such as factoid QA or knowledge base QA (Dong et al. 2015) by reference to open texts, aiming at avoiding efforts on pre-engineering and retrieving facts from a structured manual-crafted knowledge corpus.

Therefore, though MRC tasks employ the form of question answering, it can be regarded as not only just the extension or variant of QA but also a new concept concerning studying the capacity of language understanding over some context. Reading comprehension is an old term to measure the knowledge accrued through reading. When it comes to machines, it concerns that machine is trained to read unstructured natural language texts, such as a book or a news article, comprehend and absorb the knowledge without the need of human curation.

To some extent, traditional language understanding and inference tasks, such as textual entailment (TE), can be regarded as a type of MRC in theory as well. The common goal is to give a prediction after reading and comprehending the input texts; thus the NLI and standard MRC tasks are often evaluated together for assessing model's language understanding capacity (Peters et al. 2018; Radford et al. 2018; Zhang et al. 2019e, 2020b). Besides, their forms can be converted to each other. MRC can be formed as NLI format (Zhang et al. 2019b), and NLI can also be regarded as multi-choice MRC (*entailment*, *neutral*, or *contradictory*).

3.3 Task Formulation

Given the reference document or passage, as the standard form, MRC requires the machine to answer questions about it. The formation of MRC can be described as a tuple $< P, Q, A >$, where P is a passage (context), and Q is a query over the contents of P, in which A is the answer or candidate option.

Table 5: Examples of typical MRC forms.

Cloze-style	from CNN (Hermann et al. 2015)
Context	*(@entity0) – a bus carrying members of a @entity5 unit overturned at an @entity7 military base sunday , leaving 23 @entity8 injured , four of them critically , the military said in a news release . a bus overturned sunday in @entity7 , injuring 23 @entity8 , the military said . the passengers , members of @entity13 , @entity14 , @entity15 , had been taking part in a training exercise at @entity19 , an @entity21 post outside @entity22 , @entity7 . they were departing the range at 9:20 a.m. when the accident occurred . the unit is made up of reservists from @entity27 , @entity28 , and @entity29 , @entity7 . the injured were from @entity30 and @entity31 out of @entity29 , a @entity32 suburb . by mid-afternoon , 11 of the injured had been released to their unit from the hospital . pictures of the wreck were provided to the news media by the military . @entity22 is about 175 miles south of @entity32 . e-mail to a friend*
Question	*bus carrying @entity5 unit overturned at _____ military base*
Answer	*@entity7*
Multi-choice	from RACE (Lai et al. 2017)
Context	*Runners in a relay race pass a stick in one direction. However, merchants passed silk, gold, fruit, and glass along the Silk Road in more than one direction. They earned their living by traveling the famous Silk Road. The Silk Road was not a simple trading network. It passed through thousands of cities and towns. It started from eastern China, across Central Asia and the Middle East, and ended in the Mediterranean Sea. It was used from about 200 B, C, to about A, D, 1300, when sea travel offered new routes, It was sometimes called the worldâĂŹs longest highway. However, the Silk Road was made up of many routes, not one smooth path. They passed through what are now 18 countries. The routes crossed mountains and deserts and had many dangers of hot sun, deep snow, and even battles. Only experienced traders could return safely.*
Question	*The Silk Road became less important because _____.*
Answer	*A.it was made up of different routes* *B.silk trading became less popular* ***C.sea travel provided easier routes*** *D.people needed fewer foreign goods*
Span Extraction	from SQuAD (Rajpurkar et al. 2016)
Context	*Robotics is an interdisciplinary branch of engineering and science that includes mechanical engineering, electrical engineering, computer science, and others. Robotics deals with the design, construction, operation, and use of robots, as well as computer systems for their control, sensory feedback, and information processing. These technologies are used to develop machines that can substitute for humans. Robots can be used in any situation and for any purpose, but today many are used in dangerous environments (including bomb detection and de-activation), manufacturing processes, or where humans cannot survive. Robots can take on any form, but some are made to resemble humans in appearance. This is said to help in the acceptance of a robot in certain replicative behaviors usually performed by people. Such robots attempt to replicate walking, lifting, speech, cognition, and basically anything a human can do.*
Question	*What do robots that resemble humans attempt to do?*
Answer	*replicate walking, lifting, speech, cognition*
Free-form	from DROP (Dua et al. 2019)
Context	*The Miami Dolphins came off of a 0-3 start and tried to rebound against the Buffalo Bills. After a scoreless first quarter the Dolphins rallied quick with a 23-yard interception return for a touchdown by rookie Vontae Davis and a 1-yard touchdown run by Ronnie Brown along with a 33-yard field goal by Dan Carpenter making the halftime score 17-3. Miami would continue with a Chad Henne touchdown pass to Brian Hartline and a 1-yard touchdown run by Ricky Williams. Trent Edwards would hit Josh Reed for a 3-yard touchdown but Miami ended the game with a 1-yard touchdown run by Ronnie Brown. The Dolphins won the game 38-10 as the team improved to 1-3. Chad Henne made his first NFL start and threw for 115 yards and a touchdown.*
Question	*How many more points did the Dolphins score compare to the Bills by the game's end?*
Answer	*28*

In the exploration of MRC, constructing a high-quality, large-scale dataset is as important as optimizing the model structure. Following Chen (2018),[5] the existing MRC

5 We made slight modifications to adapt to the latest emerging types.

variations can be roughly divided into four categories, 1) *cloze-style*; 2) *multi-choice*; 3) *span extraction*, and 4) *free-form prediction*.

3.4 Typical Datasets

Cloze-style. For cloze-style MRC, the question contains a placeholder and the machine must decide which word or entity is the most suitable option. The standard datasets are CNN/Daily Mail (Hermann et al. 2015), Children's Book Test dataset (CBT) (Hill et al. 2015), BookTest (Bajgar, Kadlec, and Kleindienst 2016), Who did What (Onishi et al. 2016), ROCStories (Mostafazadeh et al. 2016), CliCR (Suster and Daelemans 2018).

Multi-choice. This type of MRC requires the machine to find the only correct option in the given candidate choices based on the given passage. The major datasets are MCTest (Richardson, Burges, and Renshaw 2013), QA4MRE (Sutcliffe et al. 2013), RACE (Lai et al. 2017), ARC (Clark et al. 2018), SWAG (Zellers et al. 2018), DREAM (Sun et al. 2019a), etc.

Span Extraction. The answers in this category of MRC are spans extracted from the given passage texts. The typical benchmark datasets are SQuAD (Rajpurkar et al. 2016), TrivialQA (Joshi et al. 2017), SQuAD 2.0 (extractive with unanswerable questions) (Rajpurkar, Jia, and Liang 2018), NewsQA (Trischler et al. 2017), SearchQA (Dunn et al. 2017), etc.

Free-form Prediction. The answers in this type are abstractive free-form based on the understanding of the passage. The forms are diverse, including generated text spans, yes/no judgment, counting, and enumeration. For free-form QA, the widely-used datasets are MS MACRO (Bajaj et al. 2016), NarrativeQA (Kočiskỳ et al. 2018), Dureader (He et al. 2018). This category also includes recent conversational MRC, such as CoQA (Reddy, Chen, and Manning 2019) and QuAC (Choi et al. 2018), and discrete reasoning types involving counting and arithmetic expression as those in DROP (Dua et al. 2019), etc.

Except for the variety of formats, the datasets also differ from 1) context styles, e.g., single paragraph, multiple paragraphs, long document, and conversation history; 2) question types, e.g., open natural question, cloze-style fill-in-blank, and search queries; 3) answer forms, e.g., entity, phrase, choice, and free-form texts; 4) domains, e.g., Wikipedia articles, news, examinations, clinical, movie scripts, and scientific texts; 5) specific skill objectives, e.g., unanswerable question verification, multi-turn conversation, multi-hop reasoning, mathematical prediction, commonsense reasoning, coreference resolution. A detailed comparison of the existing dataset is listed in Appendix §7.

3.5 Evaluation Metrics

For cloze-style and multi-choice MRC, the common evaluation metric is accuracy. For span-based QA, the widely-used metrics are Exact match (EM) and (Macro-averaged) F1 score. EM measures the ratio of predictions that match any one of the ground truth answers exactly. F1 score measures the average overlap between the prediction and ground truth answers. For non-extractive forms, such as generative QA, answers are not limited to the original context, so ROUGE-L (Lin 2004) and BLEU (Papineni et al. 2002) are also further adopted for evaluation.

3.6 Towards Prosperous MRC

Most recent MRC test evaluations are based on an online server, which requires to submit the model to assess the performance on the hidden test sets. Official leaderboards are also available for easy comparison of submissions. A typical example is SQuAD. [6] Open and easy following stimulate the prosperity of MRC studies, which can provide a great precedent for other NLP tasks. We think the success of the MRC task can be summarized as follows:

- **Computable Definition**: due to the vagueness and complexity of natural language, on the one hand, a clear and computable definition is essential (e.g., cloze-style, multi-choice, span-based, etc.);
- **Convincing Benchmarks**: to promote the progress of any application, technology, open, and comparable assessments are indispensable, including convincing evaluation metrics (e.g., EM and F1), and evaluation platforms (e.g., leaderboards, automatic online evaluations).

The definition of a task is closely related to the automatic evaluation. Without computable definitions, there will be no credible evaluation.

3.7 Related Surveys

Previous survey papers (Zhang et al. 2019b; Qiu et al. 2019a; Liu et al. 2019b) mainly outlined the existing corpus and models for MRC. Our survey differs from previous surveys in several aspects:

- Our work goes much deeper to provide a comprehensive and comparative review with an in-depth explanation over the origin and the development of MRC in the broader view of the NLP scenario, paying special focus on the role of CLMs. We conclude that MRC boosts the progress from language processing to understanding, and the theme of MRC is gradually moving from shallow text matching to cognitive reasoning.
- For the technique side, we propose new taxonomies of the architecture of MRC, by formulating MRC systems as two-stage architecture motivated by cognition psychology and provide a comprehensive discussion of technical methods. We summarize the technical methods and highlights in different stages of MRC development. We show that the rapid improvement of MRC systems greatly benefits from the progress of CLMs.
- Besides a wide coverage of topics in MRC researches through investigating typical models and trends from MRC leaderboards, our own empirical analysis is also provided. A variety of newly emerged topics, e.g., interpretation of models and datasets, decomposition of prerequisite skills, complex reasoning, low-resource MRC, etc., are also discussed in depth. According to our experience, we demonstrate our observations and suggestions for the MRC researches.[7]

6 https://rajpurkar.github.io/SQuAD-explorer/.

7 We are among the pioneers to research neural machine reading comprehension. We pioneered the research direction of employing linguistic knowledge for building MRC models, including morphological segmentation (Zhang, Huang, and Zhao 2018; Zhang et al. 2019e, 2018b), semantics injection (Zhang et al. 2019d, 2020b), syntactic guidance (Zhang et al. 2020c), and commonsense (Li, Zhang, and Zhao 2020). Besides the encoder representation, we investigated the decoder part to strengthen the comprehension, including interactive matching (Zhang et al. 2020a; Zhu, Zhao, and Li

We believe that this survey would help the audience more deeply understand the development and highlights of MRC, as well as the relationship between MRC and the broader NLP community.

4. Technical Methods

4.1 Two-stage Solving Architecture

Inspired by dual process theory of cognition psychology (Wason and Evans 1974; Evans 1984, 2003; Kahneman 2011; Evans 2017; Ding et al. 2019), the cognitive process of human brains potentially involves two distinct types of procedures: contextualized perception (*reading*) and analytic cognition (*comprehension*), where the former gather information in an implicit process, then the latter conduct the controlled reasoning and execute goals. Based on the above theoretical basis, in the view of architecture design, a standard reading system (reader) which solves MRC problem generally consists of two modules or building steps:

1) building a CLM as Encoder;
2) designing ingenious mechanisms as Decoder according to task characteristics.

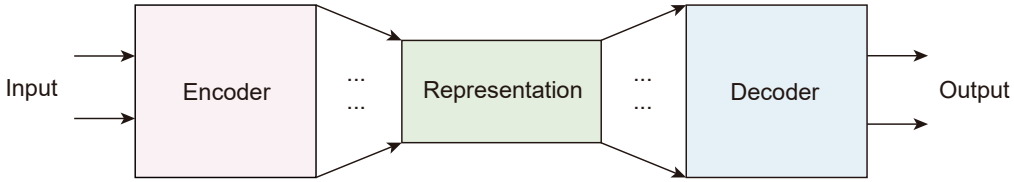

Figure 7: Encoder-Decoder Solving Architecture.

We find that the generic architecture of MRC system can thus be minimized as the formulation as two-stage solving architecture in the perspective of Encoder-Decoder architecture (Sutskever, Vinyals, and Le 2014).[8] General Encoder is to encoder the inputs as contextualized vectors, and Decoder is specific to the detailed task. Figure 7 shows the architecture.

4.2 Typical MRC Architecture

Here we introduce two typical MRC architectures following the above Encoder-Decoder framework, 1) traditional RNN-based *BiDAF* and 2) CLM-powered *BERT*.

4.2.1 Traditional RNN-based BiDAF. Before the invention of CLMs, early studies widely adopted RNNs as feature encoders for sequences, among which GRU was the most popular due to the fast and effective performance. The input parts, e.g., passage and question, are fed to the encoder separately. Then, the encoded sequences are passed

2020), answer verification (Zhang, Yang, and Zhao 2020), and semantic reasoning (Zhang, Zhao, and Zhou 2020). Our researches cover the main topics of MRC. The approaches enable effective and interpretable solutions for real-world applications, such as question answering (Zhang and Zhao 2018), dialogue and interactive systems (Zhang et al. 2018c; Zhu et al. 2018b; Zhang, Huang, and Zhao 2019). We also won various first places in major MRC shared tasks and leaderboards, including CMRC-2017, SQuAD 2.0, RACE, SNLI, and DREAM.

8 We find that most NLP systems can be formed as such architecture.

to attention layers for matching interaction between passage and questions before predicting the answers. The typical MRC model is BiDAF, which is composed of four main layers: 1) encoding layer that transforms texts into a joint representation of the word and character embeddings; 2) contextual encoding that employs BiGRUs to obtain contextualized sentence-level representation;[9]; 3) attention layer to model the semantic interactions between passage and question; 4) answer prediction layer to produce the answer. The first two layers are the counterpart of Encoder, and the last two layers serve the role of Decoder.

4.2.2 Pre-trained CLMs for Fine-tuning. When using CLMs, the input passage and question are concatenated as a long sequence to feed CLMs, which merges the encoding and interaction process in RNN-based MRC models. Therefore, the general encoder has been well formalized as CLMs, appended with a simple task-specific linear layer as Decoder to predict the answer.

4.3 Encoder

The encoder part plays the role of vectorizing the natural language texts into latent space and further models the contextualized features of the whole sequence.

4.3.1 Multiple Granularity Features.

Language Units. Utilizing fine-grained features of words was one of the hot topics in previous studies. To solve the out-of-vocabulary (OOV) problem, character-level embedding was once a common unit besides word embeddings (Seo et al. 2017; Yang et al. 2017a; Dhingra et al. 2017; Zhang et al. 2018b; Zhang, Huang, and Zhao 2019). However, character is not the natural minimum linguistic unit, which makes it quite valuable to explore the potential unit (subword) between character and word to model sub-word morphologies or lexical semantics. To take advantage of both word-level and character representations, subword-level representations for MRC were also investigated (Zhang, Huang, and Zhao 2018; Zhang et al. 2019e). In Zhang, Huang, and Zhao (2018), we propose BPE-based subword segmentation to alleviate OOV issues, and further adopt a frequency-based filtering method to strengthen the training of low-frequency words. Due to the highly flexible grained representation between character and word, subword as a basic and effective language modeling unit has been widely used for recent dominant models (Devlin et al. 2018).

Salient Features. Linguistic features, such as part-of-speech (POS) and named entity (NE) tags, are widely used for enriching the word embedding (Liu et al. 2018). Some semantic features like semantic role labeling (SRL) tags and syntactic structures also show effectiveness for language understanding tasks like MRC (Zhang et al. 2020b,c). Besides, the indicator feature, like the binary Exact Match (EM) feature is also simple and effective indications, which measures whether a context word is in the question (Chen et al. 2019).

9 Note that BiDAF has the completely contextualized encoding module. Except for the specific module implementation, the major difference with CLMs is that the BIDAF encoder is not pre-trained.

4.3.2 Structured Knowledge Injection. Incorporating human knowledge into neural models is one of the primary research interests of artificial intelligence. Recent Transformer-based deep contextual language representation models have been widely used for learning universal language representations from large amounts of unlabeled data, achieving dominant results in a series of NLU benchmarks (Peters et al. 2018; Radford et al. 2018; Devlin et al. 2018; Yang et al. 2019c; Liu et al. 2019c; Lan et al. 2019). However, they only learn from plain context-sensitive features such as character or word embeddings, with little consideration of explicit hierarchical structures that exhibited in human languages, which can provide rich dependency hints for language representation. Recent studies show that modeling structured knowledge has shown beneficial for language encoding, which can be categorized into *Linguistic Knowledge* and *Commonsense*.

Linguistic Knowledge. Language linguistics is the product of human intelligence, comprehensive modeling of syntax, semantics, and grammar is essential to provide effective structured information for effective language modeling and understanding (Zhang et al. 2020b,c, 2019d; Zhou, Zhang, and Zhao 2019).

Commonsense. At present, reading comprehension is still based on shallow segment extraction, semantic matching in limited text, and lack of modeling representation of commonsense knowledge. Human beings have learned commonsense through the accumulation of knowledge over many years. In the eyes of human beings, it is straightforward that "the sun rises in the east and sets in the west", but it is challenging to learn by machine. Commonsense tasks and datasets were proposed to facilitate the research, such as ROCStories (Mostafazadeh et al. 2016), SWAG (Zellers et al. 2018), CommonsenseQA (Talmor et al. 2019), ReCoRD (Zhang et al. 2018a), and Cosmos QA (Huang et al. 2019). Several commonsense knowledge graphs are available as the prior knowledge sources, including ConceptNet (Speer, Chin, and Havasi 2017), WebChild (Tandon, De Melo, and Weikum 2017) and ATOMIC (Sap et al. 2019). It is an important research topic to let machines learn and understand human commonsense effectively to be used in induction, reasoning, planning, and prediction.

4.3.3 Contextualized Sentence Representation. Previously, RNNs, such as LSTM, and GRU were seen as the best choice in sequence modeling or language models. However, the recurrent architectures have a fatal flaw, which is hard to parallel in the training process, limiting the computational efficiency. Vaswani et al. (2017) proposed Transformer, based entirely on self-attention rather than RNN or Convolution. Transformer can not only achieve parallel calculations but also capture the semantic correlation of any span. Therefore, more and more language models tend to choose it to be the feature extractor. Pre-trained on a large-scale textual corpus, these CLMs well serve as the powerful encoders for capturing contextualized sentence representation.

4.4 Decoder

After encoding the input sequences, the decoder part is used for solving the task with the contextualized sequence representation, which is specific to the detailed task requirements. For example, the decoder is required to select a proper question for multi-choice MRC or predict an answer span for span-based MRC.

Not until recently keep the primary focuses of nearly all MRC systems on the encoder side, i.e., the deep pre-trained models (Devlin et al. 2018), as the systems may

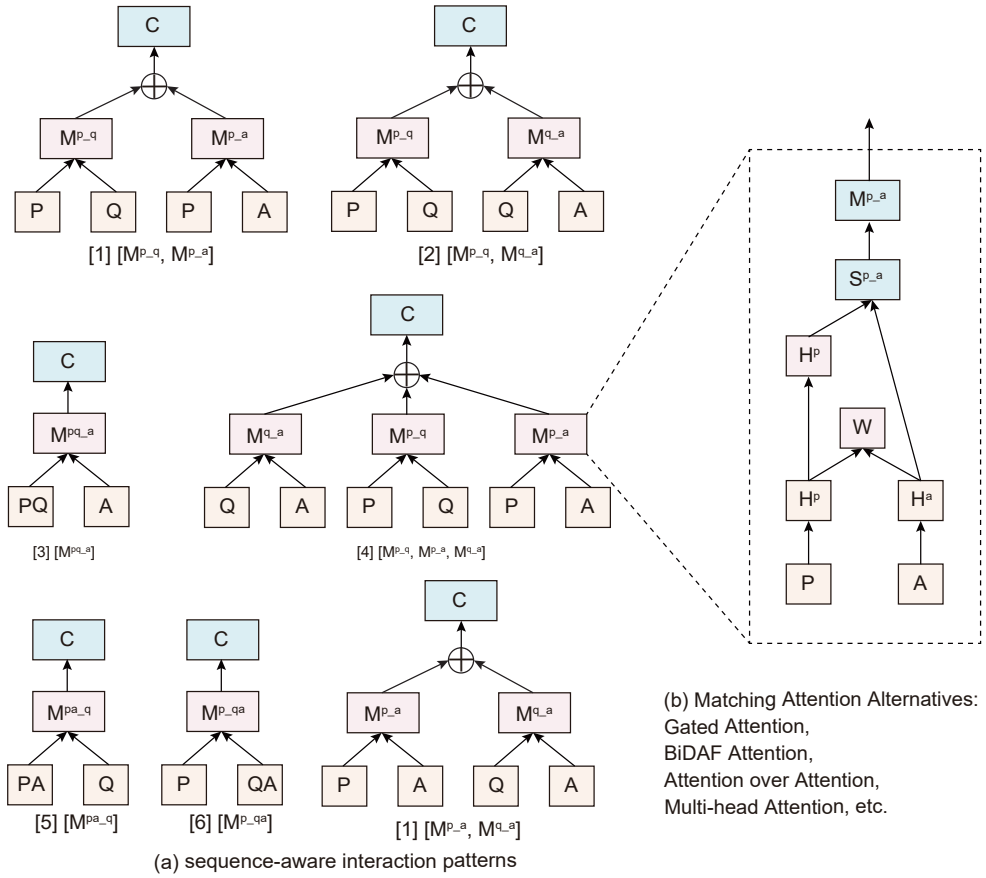

Figure 8: Designs of matching network.

simply and straightforwardly benefit from a strong enough encoder. Meanwhile, little attention is paid to the decoder side of MRC models (Hu et al. 2019c; Back et al. 2020), though it has been shown that better decoder or better manner of using encoder still has a significant impact on MRC performance, no matter how strong the encoder (i.e., the adopted pre-trained CLM) it is (Zhang et al. 2020a). In this part, we discuss the decoder design in three aspects: 1) *matching network*; 2) *answer pointer*, 2) *answer verifier*, and 3) *answer type predictor*.

4.4.1 Matching Network. The early trend is a variety of attention-based interactions between passage and question, including: Attention Sum (Kadlec et al. 2016), Gated Attention (Dhingra et al. 2017), Self-matching (Wang et al. 2017), BiDAF Attention (Seo et al. 2017), Attention over Attention (Cui et al. 2017), and Co-match Attention (Wang et al. 2018a).

Some work is also investigating the attention-based interactions of passage and question in the era of Transformer-based backbones, such as dual co-match attention (Zhang et al. 2020a; Zhu, Zhao, and Li 2020). Figure 8 presents the exhaustive patterns of matching considering three possible sequences: passage (P), question (Q), and answer

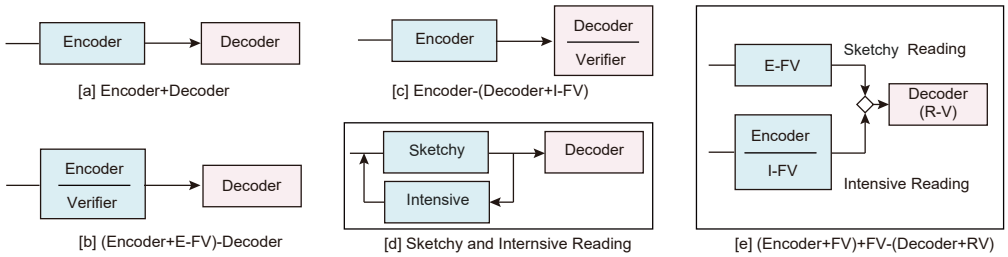

Figure 9: Designs of answer verifier.

candidate option (A).[10] The sequences, P, Q or A, can be concatenated together as one, for example, PQ denotes the concatenation of P and Q. M is defined as the matching operation. For example, M^{p_a} models the matching between the hidden states of P and A. We depict the simple but widely-used matching attention M in Figure 8-(b) for example, whose formulation is further described in §5.6.3 for detailed reference. However, the study of the matching mechanisms has come to a bottleneck facing the already powerful CLM encoders, which are essentially interactive to model paired sequences.

4.4.2 Answer Pointer. Span prediction is one of the major focuses of MRC tasks. Most models predict the answer by generating the start position and the end position corresponding to the estimated answer span. Pointer network (Vinyals, Fortunato, and Jaitly 2015) was used in early MRC models (Wang and Jiang 2016; Wang et al. 2017).

For training the model to predict the answer span for an MRC task, standard maximum-likelihood method is used for predicting exactly-matched (EM) start and end positions for an answer span. It is a strict objective that encourages exact answers at the cost of penalizing nearby or overlapping answers that are sometimes equally accurate. To alleviate the issue and predict more acceptable answers, reinforcement learning algorithm based self-critical policy learning was adopted to measure the reward as word overlap between the predicted answer and the ground truth, so as to optimize towards the F1 metric instead of EM metric for span-based MRC (Xiong, Zhong, and Socher 2018; Hu et al. 2018).

4.4.3 Answer Verifier. For the concerned MRC challenge with unanswerable questions, a reader has to handle two aspects carefully: 1) give the accurate answers for answerable questions; 2) effectively distinguish the unanswerable questions, and then refuse to answer. Such requirements complicate the reader's design by introducing an extra verifier module or answer-verification mechanism. Figure 9 shows the possible designs of the verifiers. The variants are mainly three folds (the formulations are elaborated in §5.6):

1) Threshold-based answerable verification (TAV). The verification mechanism can be simplified an answerable threshold over predicted span probability that is broadly

10 Though many well-known matching methods only involve passage and question as for cloze-style and span-based MRC, we present a more general demonstration by also considering multi-choice types that have three types of input, and the former types are also included as counterparts.

Table 6: Loss functions for MRC. CE: categorical crossentropy, BCE: binary crossentropy, MSE: mean squared error.

Type	CE	BCE	MSE
Cloze-style	✓		
Span-based	✓		
+ (binary) verification	✓	✓	✓
+ yes/no	✓	✓	✓
+ count	✓		
Multi-choice	✓		

used by powerful enough CLMs for quickly building readers (Devlin et al. 2018; Zhang et al. 2020b).

2) Multitask-style verification (Intensive). Mostly, for module design, the answer span prediction and answer verification are trained jointly with multitask learning (Figure 9(c)). Liu et al. (2018) appended an empty word token to the context and added a simple classification layer to the reader. Hu et al. (2019c) used two types of auxiliary loss, independent span loss to predict plausible answers and independent no-answer loss to decide the answerability of the question. Further, an extra verifier is adopted to decide whether the predicted answer is entailed by the input snippets (Figure 9(b)). Back et al. (2020) developed an attention-based satisfaction score to compare question embeddings with the candidate answer embeddings. It allows explaining why a question is classified as unanswerable by showing unmet conditions within the question (Figure 9(c)). Zhang et al. (2020c) proposed a linear verifier layer to context embedding weighted by start and end distribution over the context words representations concatenated to special pooled `[CLS]` token representation for BERT (Figure 9(c)).

3) External parallel verification (Sketchy). Zhang, Yang, and Zhao (2020) proposed a Retro-Reader that integrates two stages of reading and verification strategies: 1) sketchy reading that briefly touches the relationship of passage and question, and yields an initial judgment; 2) intensive reading that verifies the answer and gives the final prediction (Figure 9(d)). In the implementation, the model is structured as a rear verification (RV) method that combines the multitask-style verification as internal verification (IV), and external verification (EV) from a parallel module trained only for answerability decision, which is both simple and practicable with basically the same performance, which results in a parallel reading module design at last as the model shown in Figure 9(e).

4.4.4 Answer Type Predictor. Most of the neural reading models (Seo et al. 2017; Wang et al. 2017; Yu et al. 2018) are usually designed to extract a continuous span of text as the answer. For more open and realistic scenarios, where answers are involved with various types, such as numbers, dates, or text strings, several pre-defined modules are used to handle different kinds of answers (Dua et al. 2019; Gupta et al. 2019; Hu et al. 2019a).

4.5 Training Objectives

Table 6 shows the training objectives for different types of MRC. The widely-used objective function is cross-entropy. For some specific types, such as binary answer verification, categorical crossentropy, binary crossentropy, and mean squared error are

Table 7: Typical MRC models for comparison of Encoders on SQuAD 1.1 leaderboard. TRFM is short for Transformer. Although MRC models often employ ensembles for better performance, the results are based single models to avoid extra influence in ensemble models. * QANet and BERT used back translation and TriviaQA dataset (Joshi et al. 2017) for further data augmentation, respectively. The improvements ↑ are calculated based on the result (*italic*) on Match-LSTM.

Models	**Encoder**	**EM**	**F1**	**↑ EM**	**↑ F1**
Human (Rajpurkar, Jia, and Liang 2018)	-	82.304	91.221	-	-
Match-LSTM (Wang and Jiang 2016)	RNN	*64.744*	*73.743*	-	-
DCN (Xiong, Zhong, and Socher 2016)	RNN	66.233	75.896	1.489	2.153
Bi-DAF (Seo et al. 2017)	RNN	67.974	77.323	3.230	3.580
Mnemonic Reader (Hu, Peng, and Qiu 2017)	RNN	70.995	80.146	6.251	6.403
Document Reader (Chen et al. 2017)	RNN	70.733	79.353	5.989	5.610
DCN+ (Xiong, Zhong, and Socher 2017)	RNN	75.087	83.081	10.343	9.338
r-net (Wang et al. 2017)	RNN	76.461	84.265	11.717	10.522
MEMEN (Pan et al. 2017)	RNN	78.234	85.344	13.490	11.601
QANet (Yu et al. 2018)*	TRFM	80.929	87.773	16.185	14.030
CLMs					
ELMo (Peters et al. 2018)	RNN	78.580	85.833	13.836	12.090
BERT (Devlin et al. 2018)*	TRFM	85.083	91.835	20.339	18.092
SpanBERT (Joshi et al. 2020)	TRFM	88.839	94.635	24.095	20.892
XLNet (Yang et al. 2019c)	TRFM-XL	89.898	95.080	25.154	21.337

Table 8: Typical MRC models for comparison of Encoders on SQuAD 2.0 and RACE leaderboard. TRFM is short for Transformer. The D-values ↑ are calculated based on the results (*italic*) on BERT for SQuAD 2.0 and GTP_{v1} for RACE.

Models	**Encoder**	**SQuAD 2.0**	**↑ F1**	**RACE**	**↑ Acc**
Human (Rajpurkar, Jia, and Liang 2018)	-	91.221	-	-	
GPT_{v1} (Radford et al. 2018)	TRFM	-	-	*59.0*	-
BERT (Devlin et al. 2018)	TRFM	*83.061*	-	72.0	-
SemBERT (Zhang et al. 2020b)	TRFM	87.864	4.803	-	-
SG-Net (Zhang et al. 2020c)	TRFM	87.926	4.865	-	-
RoBERTa (Liu et al. 2019c)	TRFM	89.795	6.734	83.2	24.2
ALBERT (Lan et al. 2019)	TRFM	90.902	7.841	86.5	27.5
XLNet (Yang et al. 2019c)	TRFM-XL	90.689	7.628	81.8	22.8
ELECTRA (Clark et al. 2019c)	TRFM	91.365	8.304	-	-

also investigated (Zhang, Yang, and Zhao 2020). Similarly, for tasks involve yes or no answers, the three alternative functions are also available. For counting, previous researches tend to model it as multi-class classification task using crossentropy (Dua et al. 2019; Hu et al. 2019a; Ran et al. 2019b).

5. Technical Highlights

In this part, we summarize the previous and recent dominant techniques by reviewing the systems for the flagship datasets concerning the main types of MRC, cloze-type CNN/DailyMail (Hermann et al. 2015), multi-choice RACE (Lai et al. 2017), and span extraction SQuAD (Rajpurkar et al. 2016; Rajpurkar, Jia, and Liang 2018). Tables 7,8,9,10,11 show the statistics, from which we summarize the following observations and thoughts (we will elaborate the details in the subsequent sections):

1) **CLMs greatly boost the benchmark of current MRC**. Deeper, wider encoders carrying large-scale knowledge become a new major theme. The upper bound of the encoding capacity of deep neural networks has not been reached yet; however, training such CLMs are very time-consuming and computationally expensive. Light and refined CLMs would be more friendly for real-world and common usage, which can be realized by designing more ingenious models and learning strategies (Lan et al. 2019), as well as knowledge distillation (Jiao et al. 2019; Sanh et al. 2019).

2) **Recent years witness a decline of matching networks**. Early years witnessed a proliferation of attention-based mechanisms to improve the interaction and matching information between passage and questions, which work well with RNN encoders. After the popularity of CLMs, the advantage disappeared. Intuitively, the reason might be that CLMs are interaction-based models (e.g., taking paired sequences as input to model the interactions), but not good feature extractors. This difference might be the pre-training genre of CLMs, and also potentially due to the transformer architecture. It is also inspiring that it promotes a transformation from shallow text matching into a more complex knowledge understanding of MRC researches to some extent.

3) **Besides the encoding sides, optimizing the decoder modules is also essential for more accurate answers**. Especially for SQuAD2.0 that requires the model to decide if a question is answerable, training a separate verifier or multitasking with verification loss generally works.[11]

4) **Data augmentation from similar MRC datasets sometimes works**. Besides some work reported using TraiviaQA (Joshi et al. 2017) or NewsQA (Joshi et al. 2017) datasets as extra training data, there were also many submissions whose names contain terms about data augmentation. Similarly, when it comes to the CLMs realm, there is rarely work that uses augmentation. Besides, the pre-training of CLMs can also be regarded as data augmentation, which is highly potential for the performance gains.

In the following part, we will elaborate on the major highlights of the previous work. We also conduct a series of empirical studies to assess simple tactic optimizations as a reference for interested readers (§5.6).

5.1 Reading Strategy

Insights on the solutions to MRC challenges can be drawn from the cognitive process of humans. Therefore, some interesting reading strategies are proposed based on human reading patterns, such as Learning to Skim Text (Yu, Lee, and Le 2017), learning to stop reading (Shen et al. 2017), and our proposed retrospective reading (Zhang, Yang, and Zhao 2020). Also, (Sun et al. 2019b) proposed three general strategies: back and forth reading, highlighting, and self-assessment to improve non-extractive MRC.

11 We notice that jointly multitasking verification loss and answer span loss has been integrated as a standard module in the released codes in XLNet and ELECTRA for SQuAD2.0.

Table 9: The contributions of CLMs. * indicates results that depend on additional external training data. † indicate the result is from Yang et al. (2019c) as it was not reported in the original paper (Devlin et al. 2018). Since the final results were reported by the largest models, we listed the $large$ models for XLNet, BERT, RoBERTa, ELECTRA, and $xxlarge$ model for ALBERT. GPT is reported as the v1 version.

Method	Tokens	Size	Params	SQuAD1.1 Dev	SQuAD1.1 Test	SQuAD2.0 Dev	SQuAD2.0 Test	RACE
ELMo	800M	-	93.6M	85.6	85.8	-	-	-
GPT_{v1}	985M	-	85M	-	-	-	-	59.0
XLNet_{large}	33B	-	360M	94.5	95.1*	88.8	89.1*	81.8
BERT_{large}	3.3B	13GB	340M	91.1	91.8*	81.9	83.0	72.0†
RoBERTa_{large}	-	160GB	355M	94.6	-	89.4	89.8	83.2
$\text{ALBERT}_{xxlarge}$	-	157GB	235M	94.8	-	90.2	90.9	86.5
ELECTRA_{large}	33B	-	335M	94.9	-	90.6	91.4	-

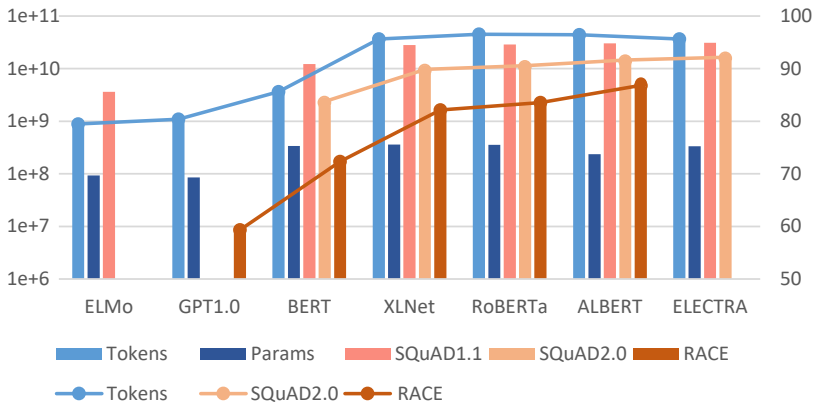

Figure 10: The contribution of the sizes of pre-trained corpus and CLMs. The right axis is the main metric for the statistics. The numbers of tokens and parameters are normalized by $\log_{10^6}(x) + 50$ where x denotes the original number. The left axis corresponds to the original values of tokens and parameters for easy reference.

5.2 CLMs Become Dominant

As shown in Table 9, CLMs improve the MRC benchmarks to a much higher stage. Besides the contextualized sentence-level representation, the advance of CLMs is also related to the much larger model size and large-scale pre-training corpus. From Table 9 and the further illustration in Figure 10, we see that both the model sizes and the scale of training data are increasing remarkably, that contribute the downstream MRC model performance.[12]

12 The influence of model parameters can also be easily verified at the SNLI leaderboard: https://nlp.stanford.edu/projects/snli/.

Table 10: Typical MRC models for comparisons of decoding designs on multi-choice RACE test sets. The matching patterns correspond to those notations in Figure 8. M: RACE-M, H: RACE-H. M, H, RACE are the accuracy on two subsets and the overall test sets, respectively.

Model	Matching	M	H	RACE
Human Ceiling Performance (Lai et al. 2017)		95.4	94.2	94.5
Amazon Mechanical Turker (Lai et al. 2017)		85.1	69.4	73.3
HAF (Zhu et al. 2018a)	$[M^{P\_A}; M^{P\_Q}; M^{Q\_A}]$	45.0	46.4	46.0
MRU (Tay, Tuan, and Hui 2018)	$[M^{P\_Q\_A}]$	57.7	47.4	50.4
HCM (Wang et al. 2018a)	$[M^{P\_Q}; M^{P\_A}]$	55.8	48.2	50.4
MMN (Tang, Cai, and Zhuo 2019)	$[M^{Q\_A}; M^{A\_Q}; M^{P\_Q}; M^{P\_A}]$	61.1	52.2	54.7
GPT (Radford et al. 2018)	$[M^{P\_Q\_A}]$	62.9	57.4	59.0
RSM (Sun et al. 2019b)	$[M^{P\_QA}]$	69.2	61.5	63.8
DCMN (Zhang et al. 2019a)	$[M^{PQ\_A}]$	77.6	70.1	72.3
OCN (Ran et al. 2019a)	$[M^{P\_Q\_A}]$	76.7	69.6	71.7
BERT$_{large}$ (Pan et al. 2019b)	$[M^{P\_Q\_A}]$	76.6	70.1	72.0
XLNet (Yang et al. 2019c)	$[M^{P\_Q\_A}]$	85.5	80.2	81.8
+ DCMN+ (Zhang et al. 2020a)	$[M^{P\_Q}; M^{P\_O}; M^{Q\_O}]$	86.5	81.3	82.8
RoBERTa (Liu et al. 2019c)	$[M^{P\_Q\_A}]$	86.5	81.8	83.2
+ MMM (Jin et al. 2019a)	$[M^{P\_Q\_A}]$	89.1	83.3	85.0
ALBERT (Jin et al. 2019a)	$[M^{P\_Q\_A}]$	89.0	85.5	86.5
+ DUMA (Zhu, Zhao, and Li 2020)	$[M^{P\_QA}; M^{QA\_P}]$	90.9	86.7	88.0
Megatron-BERT (Shoeybi et al. 2019)	$[M^{P\_Q\_A}]$	91.8	88.6	89.5

5.3 Data Augmentation

Since most high-quality MRC datasets are human-annotated and inevitably relatively small, another simple method to boost performance is data augmentation. Early effective data augmentation is to inject extra similar MRC data for training a specific model. Recently, using CLMs, which pre-trained on large-scale unlabeled corpora, can be regarded as a kind of data augmentation as well.

Training Data Augmentation. There are various methods to provide extra data to train a more powerful MRC model, including: 1) Combining various MRC datasets as training data augmentation (TDA) (Yang et al. 2019a,b); 2) Multi-tasking (Xu et al. 2018; Fisch et al. 2019); 3) Automatic question generation, such as back translation (Yu et al. 2018) and synthetic generation (Du, Shao, and Cardie 2017; Du and Cardie 2017; Kim et al. 2019; Zhu et al. 2019; Alberti et al. 2019). However, we find the gains become small when using CLMs, which might already contain the most common and important knowledge between different datasets.

Large-scale Pre-training. Recent studies showed that CLMs well acquired linguistic information through pre-training (Clark et al. 2019b; Ettinger 2020) (more discussions in Section §6.1), which is potential to the impressive results on MRC tasks.

Table 11: Results on cloze CNN/DailyMail test sets. UA: unidirectional attention. BA: bidirectional attention. The statistics are from Seo et al. (2017).

Method	Att. Type	CNN		DailyMail	
		val	test	val	test
Attentive Reader (Hermann et al. 2015)	UA	61.6	63.0	70.5	69.0
AS Reader (Kadlec et al. 2016)	UA	68.6	69.5	75.0	73.9
Iterative Attention (Sordoni et al. 2016)	UA	72.6	73.3	-	-
Stanford AR (Chen, Bolton, and Manning 2016)	UA	73.8	73.6	77.6	76.6
GAReader (Dhingra et al. 2017)	UA	73.0	73.8	76.7	75.7
AoA Reader (Cui et al. 2017)	BA	73.1	74.4	-	-
BiDAF (Seo et al. 2017)	BA	76.3	76.9	80.3	79.6

5.4 Decline of Matching Attention

As the results shown in Tables 10-11, it is easy to notice that the attention mechanism is the key component in previous RNN-based MRC systems.[13]

We see that bidirectional attention (BA) works better than unidirectional one, and co-attention is a superior matching method, which indicate the advance of more rounds of matching that would be effective at capturing more fine-grained information intuitively. When using CLMs as the encoder, we observe that the explicit passage and question attention could only show quite marginal, or even degradation of performance. The reason might be that CLMs are interaction-based matching models (Qiao et al. 2019) when taking the whole concatenated sequences of passage and question. It is not suggested to be employed as a representative model. Bao et al. (2019) also reported similar observations, showing that the unified modeling of sequences in BERT outperforms previous networks that separately treat encoding and matching.

After contextualized encoder by the CLMs, the major connections for reading comprehension might have been well modeled, and the vital information is aggregated to the representations of special tokens, such as `[CLS]` and `[SEP]` for BERT. We find that the above encoding process of CLMs is quite different from that in traditional RNNs, where the hidden states of each token are passed successively in one direction, without mass aggregation and degradation of representations.[14] The phenomenon may explain why interactive attentions between input sequences work well with RNN-based feature extractors but show no obvious advantage in the realm of CLMs.

5.5 Tactic Optimization

The objective of answer verification. For answer verification, modeling the objective as classification or regression would have a slight influence on the final results. However, the advance might vary based on the backbone network, as some work took the regression loss due to the better performance (Yang et al. 2019c), while the recent work reported that the classification would be better in some cases (Zhang, Yang, and Zhao 2020).

13 We roughly summarize the matching methods in the previous work using our model notations, which meet their general ideas except some calculation details.

14 Although the last hidden state is usually used for the overall representation, the other states may not suffer from degradation like in multi-head attention-based deep CLMs.

The dependency inside answer span. Recent CLM-based models simplified the span prediction part as independent classification objectives. However, the end position is related to the start predictions. As a common method in early works (Seo et al. 2017), jointly integrating the start logits and the sequence hidden states to obtain the end logits is potential for further enhancement. Another neglected aspect recently is the dependence of all the tokens inside an answer span, instead of considering only the start and end positions.

Re-ranking of candidate answers. Answer reranking is adapted to mimic the process of double-checking. A simple strategy is to use N-best reranking strategy after generating answers from neural networks (Cui et al. 2017; Wang et al. 2018b,c,d; Hu et al. 2019b). Unlike previous work that ranks candidate answers, Hu et al. (2019a) proposed an arithmetic expression reranking mechanism to rank expression candidates that are decoded by beam search, to incorporate their context information during reranking to confirm the prediction further.

5.6 Empirical Analysis of Decoders

To gain insights on how to further improve MRC, we report our attempts to improve model performance with general and straightforward tactic optimizations for the widely-used SQuAD2.0 dataset that does not rely on the backbone model. The methods include three types, *Verification*, *Interaction*, and *Answer Dependency*.[15]

5.6.1 Baseline. We adopt BERT$_{large}$ (Devlin et al. 2018) and ALBERT$_{xxlarge}$ (Lan et al. 2019) as our baselines.

Encoding. The input sentence is first tokenized to word pieces (subword tokens). Let $T = \{t_1, \ldots, t_L\}$ denote a sequence of subword tokens of length L. For each token, the input embedding is the sum of its token embedding, position embedding, and token-type embedding. Let $X = \{x_1, \ldots, x_L\}$ be the outputs of the encoder, which are embedding features of encoding sentence words of length L. The input embeddings are then fed into the deep Transformer (Vaswani et al. 2017) layers for learning contextual representations. Let $X^g = \{x_L^g, \ldots, x_L^g\}$ be the features of the g-th layer. The features of the $g+1$-th layer, x^{g+1} is computed by

$$\tilde{h}_i^{g+1} = \sum_{m=1}^{M} W_m^{g+1} \left\{ \sum_{j=1}^{n} A_{i,j}^m \cdot V_m^{g+1} x_j^g \right\}, \tag{8}$$

$$h_i^{g+1} = \mathit{LayerNorm}(x_i^g + \tilde{h}_i^{g+1}), \tag{9}$$

$$\tilde{x}_i^{g+1} = W_2^{g+1} \cdot \mathit{GELU}(W_1^{g+1} h_i^{g+1} + b_1^{g+1}) + b_2^{g+1}, \tag{10}$$

$$x_i^{g+1} = \mathit{LayerNorm}(h_i^{g+1} + \tilde{x}_i^{g+1}), \tag{11}$$

15 In this part, we intend to intuitively show what kinds of tactic optimizations potentially work, so we brief the details of the methods and report the best results as a reference after hyper-parameter searching. We recommend interested readers to read our technical report (Zhang, Yang, and Zhao 2020) for the details of answer verification and sequence interactions. Our sources are publicly available at https://github.com/cooelf/AwesomeMRC.

where m is the index of the attention heads, and $A^m_{i,j} \propto \exp[(Q^{g+1}_m x^g_i)^\top (K^{g+1}_m x^g_j)]$ denotes the attention weights between elements i and j in the m-th head, which is normalized by $\sum_{j=1}^{N} A^m_{i,j} = 1$. $W^{g+1}_m, Q^{g+1}_m, K^{g+1}_m$ and V^{g+1}_m are learnable weights for the m-th attention head, W^{g+1}_1, W^{g+1}_2 and b^{g+1}_1, b^{g+1}_2 are learnable weights and biases, respectively. Finally, we have last-layer hidden states of the input sequence $\mathbf{H} = \{h_1, \ldots, h_L\}$ as the contextualized representation of the input the sequence.

Decoding. The aim of span-based MRC is to find a span in the passage as answer, thus we employ a linear layer with SoftMax operation and feed $\mathbf{H}$ as the input to obtain the start and end probabilities, s and e:

$$s, e \propto \mathit{SoftMax}(\mathit{Linear}(\mathbf{H})). \tag{12}$$

Threshold based answerable verification (TAV). For unanswerable question prediction, given output start and end probabilities s and e, and the verification probability v, we calculate the has-answer score $score_{has}$ and the no-answer score $score_{na}$:

$$\begin{aligned} score_{has} &= \max(s_{k_1} + e_{k_2}), 1 < k_1 \leq k_2 \leq L, \\ score_{na} &= s_1 + e_1, \end{aligned} \tag{13}$$

where s_1 and e_1 denote the corresponding logits for the special token `[CLS]` as in BERT-based models used for answer verification (Devlin et al. 2018; Lan et al. 2019). We obtain a difference score between *has-answer* score and the *no-answer* score as final score. An answerable threshold δ is set and determined according to the development set. The model predicts the answer span that gives the *has-answer* score if the final score is above the threshold δ, and null string otherwise.

Training Objective. The training objective of answer span prediction is defined as cross entropy loss for the start and end predictions,

$$\mathbb{L}^{span} = -\frac{1}{N} \sum_{i}^{N} [\log(p^s_{y^s_i}) + \log(p^e_{y^e_i})], \tag{14}$$

where y^s_i and y^e_i are respectively ground-truth start and end positions of example i. N is the number of examples.

5.6.2 Verification. Answer verification is vital for MRC tasks that involve unanswerable answers. We tried to add an external separate classifier model that is the same as the MRC model except for the training objective (E-FV). We weighted the predicted verification logits and original heuristic no-answer logits to decide whether the question is answerable. Besides, we also investigated adding multitasking the original span loss with verification loss as an internal front verifier (I-FV). The internal verification loss can be a cross-entropy loss (I-FV-CE), binary cross-entropy loss (I-FV-BE), or regression-style mean square error loss (I-FV-MSE).

The pooled first token (special symbol, [CLS]) representation $h_1 \in \mathbf{H}$, as the overall representation of the sequence,[16] is passed to a fully connection layer to get classification logits or regression score. Let $\hat{y}_i \propto \textit{Linear}(h_1)$ denote the prediction and y_i is the answerability target, the three alternative loss functions are as defined as follows:

(1) For cross entropy as loss function for the classification verification:

$$\begin{aligned} \hat{y}_{i,k} &= \textit{SoftMax}(\textit{Linear}(h_1)), \\ \mathbb{L}^{ans} &= -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\left[y_{i,k}\log\hat{y}_{i,k}\right], \end{aligned} \tag{15}$$

where K is the number of classes. In this work, $K = 2$.

(2) For binary cross entropy as loss function for the classification verification:

$$\begin{aligned} \hat{y}_i &= \textit{Sigmoid}(\textit{Linear}(h_1)), \\ \mathbb{L}^{ans} &= -\frac{1}{N}\sum_{i=1}^{N}\left[y_i\log\hat{y}_i + (1-y_i)\log(1-\hat{y}_i)\right]. \end{aligned} \tag{16}$$

(3) For the regression verification, mean square error is adopted as its loss function:

$$\begin{aligned} \hat{y}_i &= \textit{Linear}(h_1), \\ \mathbb{L}^{ans} &= \frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2. \end{aligned} \tag{17}$$

During training, the joint loss function for FV is the weighted sum of the span loss and verification loss.

$$\mathbb{L} = \alpha_1\mathbb{L}^{span} + \alpha_2\mathbb{L}^{ans}, \tag{18}$$

where α_1 and α_2 are weights. We set $\alpha_1 = \alpha_2 = 0.5$ for our experiments.

We empirically find that training with joint loss can yield better results, so we also report the results of 1) summation of all the I-FV losses (All I-FVs: I-FV-CE, I-FV-BE, and I-FV-MSE), 2) combination of external and internal verification (All I-FVs + E-FV) by calculating the sum (denoted as v) of the logits of E-FV and I-FVs as the final answerable logits. In the later scenario, the TAV is rewritten as,

$$\begin{aligned} score_{has} &= \max(s_{k_1} + e_{k_2}), 1 < k_1 \leq k_2 \leq L, \\ score_{na} &= \lambda_1(s_1 + e_1) + \lambda_2 v, \end{aligned} \tag{19}$$

where λ_1 and λ_2 are weights. We set $\lambda_1 = \lambda_2 = 0.5$.

16 Following the initial practice of BERT-tyle models, the first token (special symbol, [CLS]) representation is supposed to be the overall representation of the sequence owing to the pre-training objective.

5.6.3 Interaction. To obtain the representation of each passage and question, we split the last-layer hidden state $\mathbf{H}$ into $\mathbf{H}^Q$ and $\mathbf{H}^P$ as the representations of the question and passage, according to its position information. Both of the sequences are padded to the maximum length in a minibatch. Then, we investigate two potential question-aware matching mechanisms, 1) Transformer-style multi-head attention (MH-ATT), and 2) traditional dot attention (DT-ATT).

• **Multi-head Attention** We feed the $\mathbf{H}^Q$ and $\mathbf{H}$ to a revised one-layer multi-head attention layer inspired by Lu et al. (2019).[17] Since the setting is $\mathbf{Q} = \mathbf{K} = \mathbf{V}$ in multi-head self attention,[18] which are all derived from the input sequence, we replace the input to $\mathbf{Q}$ with $\mathbf{H}$, and both of $\mathbf{K}$ and $\mathbf{V}$ with $\mathbf{H}^Q$ to obtain the question-aware context representation $\mathbf{H}'$.

• **Dot Attention** Another alternative is to feed $\mathbf{H}^Q$ and $\mathbf{H}$ to a traditional matching attention layer (Wang et al. 2017), by taking the question presentation $\mathbf{H}^Q$ as the attention to the representation $\mathbf{H}^C$:

$$\begin{aligned} \mathbf{M} &= \mathit{SoftMax}(\mathbf{H}(\mathbf{W}_p\mathbf{H}^Q + \mathbf{b}_p \otimes \mathbf{e}_q)^\mathsf{T}), \\ \mathbf{H}' &= \mathbf{M}\mathbf{H}^Q, \end{aligned} \tag{20}$$

where $\mathbf{W}_q$ and $\mathbf{b}_q$ are learnable parameters. $\mathbf{e}_q$ is a all-ones vector and used to repeat the bias vector into the matrix. $\mathbf{M}$ denotes the weights assigned to the different hidden states in the concerned two sequences. $\mathbf{H}'$ is the weighted sum of all the hidden states and it represents how the vectors in $\mathbf{H}$ can be aligned to each hidden state in $\mathbf{H}^Q$.

Finally, the representation $\mathbf{H}'$ is used for the later predictions as described in the *decoding* and *TAV* section above.

5.6.4 Answer Dependency. Recent studies separately use $\mathbf{H}$ to predict the start and end spans for the answer, neglecting the dependency of the start and end representations. We model the dependency between start and end logits by concatenating the start logits and $\mathbf{H}$ through a linear layer to obtain the end logits:

$$e = \mathit{Linear}([s; \mathbf{H}]), \tag{21}$$

where $[;]$ denotes concatenation.

5.6.5 Findings. Table 12 shows the results. Our observations are as follows:

- For answer verification, either of the front verifiers boosts the baselines, and integrating all the verifiers can yield even better results.
- Adding extra interaction layers after the strong CLMs could only yield marginal improvement, which verifies the CLMs' strong ability to capture the relationships between passage and question.
- Answer dependency can effectively improve the exact match score, which can intuitively help yield a more exactly matched answer span.

17 We do not use $\mathbf{H}^P$ because $\mathbf{H}$ achieved better results in our preliminary experiments.
18 In this work, Q, K, V correspond to the items $Q_m^{g+1}x_i^g, K_m^{g+1}x_j^g$) and $V_m^{g+1}x_j^g$, respectively.

Table 12: Results (%) of different decoder mechanisms on the SQuAD2.0 dev set. Part of the numbers of *verification* and *interactions* are adapted from our previous work (Zhang, Yang, and Zhao 2020) (slight update with further hyperparameter tuning).

Method	BERT		ALBERT	
	EM	**F1**	**EM**	**F1**
Baseline	78.8	81.7	87.0	90.2
Interaction				
+ MH-ATT	78.8	81.7	87.3	90.3
+ DT-ATT	78.3	81.4	86.8	90.0
Verification				
+ E-FV	79.1	82.1	87.4	90.6
+ I-FV-CE	78.6	82.0	87.2	90.3
+ I-FV-BE	78.8	81.8	87.2	90.2
+ I-FV-MSE	78.5	81.7	87.3	90.4
+ All I-FVs	79.4	82.1	87.5	90.6
+ All I-FVs + E-FV	79.8	82.7	87.7	90.8
Answer Dependency				
+ SED	79.1	81.9	87.3	90.3

6. Trends and Discussions

6.1 Interpretability of Human-parity Performance

Recent years witnessed frequent reports of super human-parity results in MRC leaderboards, which further stimulated the research interests of investigating what the 'real' ability of MRC systems, and what kind of knowledge or reading comprehension skills the systems have grasped. The interpretation appeal to aspects of CLM models, MRC datasets, and models.

For CLM models. Since CLM models serve as the basic module for contextualized text representation, fingering out what the knowledge captured, especially what linguistic capacities CLMs process confer upon models, is critical for fine-tuning downstream tasks, so is for MRC. There are heated discussions about what CLM models learn recently. Recent work has tried to give the explanation by investigating the attention maps from the multi-head attention layers (Clark et al. 2019b), and conducting diagnostic tests (Ettinger 2020). Clark et al. (2019b) found that attention heads correspond well to linguistic notions of syntax and coreference. Ettinger (2020) introduced a suite of diagnostic tests to assess the linguistic competencies of BERT, indicating that BERT performs sensitivity to role reversal and same-category distinctions. Still, it struggles with challenging inferences and role-based event prediction, and it shows obvious failures with the meaning of negation.

For MRC datasets and models. So far, the MRC system is still a black box, and it is very risky to use it in many scenarios in which we have to know how and why the answer is obtained. It is critical to deeply investigate the explanation of the MRC models or design an explainable MRC architecture. Although MRC datasets are emerging rapidly and the corresponding models continuously show impressive results, it still hard to

interpret what MRC systems learned, so is the benchmark capacity of the diversity of MRC datasets (Sugawara et al. 2018, 2019; Schlegel et al. 2020). The common arguments are the overestimated ability of MRC systems as MRC models do not necessarily provide human-level understanding, due to the unprecise benchmarking on the existing datasets. Although there are many models show human-parity scores so far, we cannot say that they successfully perform human-level reading comprehension. The issue mainly lies within the low interpretability of both of the explicit internal processing of currently prevalent neural models, and what is measured by the datasets. Many questions can be answered correctly by the model that do not necessarily require grammatical and complex reasoning. For example, Jia and Liang (2017) and Wallace et al. (2019) provided manually crafted adversarial examples to show that MRC systems are easily distracted. Sugawara et al. (2019) also indicated that most of the questions already answered correctly by the model do not necessarily require grammatical and complex reasoning. The distractors can not only assess the vulnerability of the current models but also serve as salient hard negative samples to strengthen model training (Gao et al. 2019b).

Besides, as discussed in our previous work (Zhang, Yang, and Zhao 2020), since current results are relatively high in various MRC benchmark datasets, with relatively marginal improvement, it is rarely confirmed that produced results are statistically significant than baseline. For the reproducibility of models, it is necessary to conduct statistical tests in evaluating MRC models.

6.2 Decomposition of Prerequisite Skills

As the experience of human examinations, good comprehension requires different dimensions of skills. The potential solution for our researches is to decompose the skills required by the dataset and take skill-wise evaluations, thus provide more explainable and convincing benchmarking of model capacity. Further, it would be beneficial to consider following a standardized format, which can make it simpler to conduct cross-dataset evaluations (Fisch et al. 2019), and train a comprehensive model that can work on different datasets with specific skills.

Regarding the corresponding benchmark dataset construction, it is no coincidence that SQuAD datasets turned out a success and have served as the standard benchmark. Besides the high quality and specific focus of the datasets, an online evaluation platform that limits the submission frequency also ensures the convincing assessment. On the other hand, it is natural to be cautious for some comprehensive datasets with many complex question types, which requires many solver modules, as well as processing tricks–we should report convincing evaluation with detailed evaluation on separate types or subtasks, instead of just pursuing overall SOTA results. Unless honestly reporting and unifying the standards of these processing tricks, the evaluation would be troublesome and hard to replicate.

6.3 Complex Reasoning

Most of the previous advances have focused on *shallow* QA tasks that can be tackled very effectively by existing retrieval and matching-based techniques. Instead of measuring the comprehension and understanding of the QA systems in question, these tasks test merely the capability of a method to focus attention on specific words and pieces of text. To better align the progress in the field of QA with the expectations that we have of human performance and behavior when solving such tasks, a new class of questions,

e.g., “complex" or “challenge" reasoning, has been a hot topic. Complex reasoning can most generally be thought of as instances that require intelligent behavior and reasoning on the part of a machine to solve.

As the knowledge, as well as the questions themselves, become more complex and specialized, the process of understanding and answering these questions comes to resemble human expertise in specialized domains. Current examples of such complex reasoning tasks, where humans presently rule the roost, include customer support, standardized testing in education, and domain-specific consultancy services, such as medical and legal advice. The study of such complex reasoning would be promising for machine intelligence from current perception to next-stage cognition.

Recent studies have been proposed for such kind of comprehension, including multi-hop QA (Welbl, Stenetorp, and Riedel 2018; Yang et al. 2018) and conversational QA (Reddy, Chen, and Manning 2019; Choi et al. 2018). To deal with the complex multi-hop relationship, dedicated mechanism design is needed for multi-hop commonsense reasoning. Besides, structured knowledge provides a wealth of prior commonsense context, which promotes the research on the fusion of multi-hop commonsense knowledge between symbol and semantic space in recent years (Lin et al. 2019; Ma et al. 2019). For conversational QA, modeling multi-turn dependency requires extra memory designs to capture the context information flow and solve the problems precisely and consistently (Huang, Choi, and tau Yih 2019).

Regarding technical side, graph-based neural networks (GNN), including graph attention network, graph convolutional network, and graph recurrent network have been employed for complex reasoning (Song et al. 2018; Qiu et al. 2019b; Chen, Wu, and Zaki 2019; Jiang et al. 2019; Tu et al. 2019, 2020). The main intuition behind the design of GNN based models is to answer questions that require to explore and reason over multiple scattered pieces of evidence, which is similar to human’s interpretable step-by-step problem-solving behavior. Another theme appearing frequently in machine learning in general is the revisiting of the existing models and how they perform in a fair experimental setting. Shao et al. (2020) raised a concern that graph structure may not be necessary for multi-hop reasoning, and graph-attention can be considered as a particular case of self-attention as that used in CLMs. We can already see a transformation from heuristic applications of GNNs to more sound approaches and discussions about the effectiveness of graph models. For future studies, an in-depth analysis of GNNs, as well as the connections and differences between GNNs and CLMs would be inspiring.

6.4 Large-scale Comprehension

Most current MRC systems are based on the hypothesis of given passages as reference context. However, for real-world MRC applications, the reference passages, even documents, are always lengthy and detail-riddled. However, recent LM based models work slowly or even unable to process long texts. The ability of knowledge extraction is especially needed for open-domain and free-form QA whose reference texts are usually large-scale (Guu et al. 2020). A simple solution is to train a model to select the relevant information pieces by calculating the similarity with the question (Chen et al. 2017; Clark and Gardner 2018; Htut, Bowman, and Cho 2018; Tan et al. 2018; Wang et al. 2018c; Zhang, Zhao, and Zhang 2019; Yan et al. 2019; Min et al. 2018; Nishida et al. 2019). Another technique is to summarize the significant information of the reference context, by taking advantage of text summarization or compression (Li et al. 2019c).

6.5 Low-resource MRC

Low-resource processing is a hot research topic since most of the natural languages lack abundant annotated data (Wang et al. 2019; Zhang et al. 2019c). Since most MRC studies are based on the English datasets, there exists a considerable gap for other languages that do not have high-quality MRC datasets. Such a situation can be alleviated by transferring the well-trained English MRC models through domain adaptation (Wang et al. 2019), and training semi-supervised (Yang et al. 2017b; Zhang and Zhao 2018) or multilingual MRC systems (Liu et al. 2019a; Lee et al. 2019; Cui et al. 2019).

The other major drawback exposed in MRC systems is the inadequate knowledge transferability (Talmor and Berant 2019) as they are trained, and even over-fitted on specific datasets. Since most of the famous datasets are built from Wikipedia articles, the apparent benefits from CLMs might be the same or similar text patterns contained in the training corpus, e.g., context, topics, etc. It remains a significant challenge to design robust MRC models that are immune to real noise. It is also essential to build NLP systems that generalize across domains, especially unseen domains (Fisch et al. 2019).

6.6 Multimodal Semantic Grounding

Compared with human learning, the current pure text processing model performance is relatively weak, because this kind of model only learns the text features, without the perception of the external world, such as visual information. In human learning, people usually understand the world through visual images, auditory sounds, words, and other modes. Human brain perceives the world through multimodal semantic understanding. Therefore, multimodal semantic modeling is closer to human perception, which is conducive to a more comprehensive language understanding. It remains an open problem when and how to make full use of different modalities to improve reading comprehension and inference. A related research topic is visual question answering (Goyal et al. 2017), which aims to answer questions according to a given image. However, it is still in the early stage of research as the QA is concerned with only one image context. As a more practical scenario, jointly modeling diverse modalities will be potential research interests, and beneficial for real-world applications, e.g., E-commerce customer support. For example, given the mixed text, image, and audio background conversation context, the machine is required to give responses to the inquiry accordingly. With the continuous advance of computational power, we believe the joint supervision of auditory, tactile, and visual sensory information together with the language will be crucial for next-stage cognition.

6.7 Deeper But Efficient Network

Besides the high-quality benchmark datasets, the increase the computational resources, e.g., GPU, enables us to build deeper and wider networks. The last decade witnessed the traditional feature extractor from the RNN to deep transformers, with a larger capacity for contextualized modeling. In the future, we are confident that much deeper and stronger backbone frameworks will be proposed with the rapid development of GPU capacity and further boost the MRC system benchmark performance. In the meantime, smaller and refined systems, potentially through knowledge distillation from large models, also occupy a certain market, which relies on rapid and accurate reading comprehension solving ability for real-world application.

Table 1: Cloze-style MRC datasets.

Name	Size	Domain	Src	Feature
CNN/ DailyMail (Hermann et al. 2015)	1.4M	news article	A	entity cloze
Children's Book Test (Hill et al. 2015)	688K	narrative	A	large-scale automated
BookTest (Bajgar, Kadlec, and Kleindienst 2016)	14.1M	narrative	A	similar to CBT, but much larger
Who did What (Onishi et al. 2016)	200K	news article	A	cloze of person name
ROCStories (Mostafazadeh et al. 2016)	50K*5	narrative	C	Commonsense Stories
CliCR (Suster and Daelemans 2018)	100K	clinical case text	A	cloze style queries

Table 2: Multi-choice MRC datasets.

Name	Size	Domain	Src	Feature
QA4MRE (Sutcliffe et al. 2013)	240	technical document	X	exam-level questions
MCTest (Richardson, Burges, and Renshaw 2013)	2.6K	written story	C	children-level narrative
RACE (Lai et al. 2017)	100K	language exam	X	middle/high school English exam in China
Story Cloze Test (Mostafazadeh et al. 2017)	3.7K	written story	C	98,159 stories for training
TextbookQA (Kembhavi et al. 2017)	26K	textbook	X	figures involved
ARCT (Habernal et al. 2018)	2.0K	debate article	C/X	reasoning on argument
CLOTH (Xie et al. 2018)	99K	various	X	cloze exam
MCScript (Ostermann et al. 2018)	30K	written story	C	commonsense reasnoing, script knowledge
ARC (Clark et al. 2018)	8K	science exam	X	retrieved documents from textbooks
MultiRC (Khashabi et al. 2018)	6K	various documents	C	multi-sentence reasoning
SWAG (Zellers et al. 2018)	113K	video captions	M	commonsense reasoning
OpenbookQA (Mihaylov et al. 2018)	6.0K	textbook	C	commonsense reasoning
RecipeQA (Yagcioglu et al. 2018)	36K	recipe script	A	multimodal questions
Commonsense QA (Talmor et al. 2019)	12K	ConceptNet	C	commonsense reasoning
DREAM (Sun et al. 2019a)	10K	language exam	X	dialogue-based, 6.4k multi-party dialogues
MSCript 2.0 (Ostermann, Roth, and Pinkal 2019)	20K	narrative	C	commonsense reasoning, script knowledge
HellaSWAG (Zellers et al. 2019)	70K	web snippet	A	commonsense reasoning, WikiHow and ActivityNet
CosmosQA (Huang et al. 2019)	36K	narrative	C	commonsense reasoning
QuAIL (Rogers et al. 2020)	15K	various	C	prerequisite real tasks

Table 3: Span-extraction MRC datasets.

Name	Ans	Size	Domain	Src	Feature
SQuAD 1.1 (Rajpurkar et al. 2016)	Ex	100K	Wikipedia	C	large-scale crowdsourced
NewsQA (Trischler et al. 2017)	Ex	120K	news article	C	blindly created questions
SearchQA (Dunn et al. 2017)	Ex	140K	web snippet	C/X	snippets from search engine
TriviaQA (Joshi et al. 2017)	Ex	650K	web snippet	C/X	trivia questions
Quasar (Dhingra, Mazaitis, and Cohen 2017)	Ex	80K	web snippet	Q	search queries
AddSent SQuAD (Jia and Liang 2017)	Ex	3.6K	Wikipedia	C	distracting sentences injected
QAngaroo (Welbl, Stenetorp, and Riedel 2018)	Ex	50K	Wikipedia, MEDLINE	A	multi-hop reasoning
DuoRC (Saha et al. 2018)	Ex	186K	movie script	C	commonsense reasoning, multi-sentence reasoning
ProPara (Dalvi et al. 2018)	Ex	2K	science exam	A	procedural understanding
Multi-party Dialog (Ma, Jurczyk, and Choi 2018)	Ex	13K	TV show transcript	A	1.7k crowdsourced dialogues, cloze query
SQuAD 2.0 (Rajpurkar, Jia, and Liang 2018)	Ex (+NA)	100K	Wikipedia	C	unanswerable questions
Textworlds QA (Labutov et al. 2018)	Ex	1.2M	generated text	A	simulated worlds, logical reasoning
emrQA (Pampari et al. 2018)	Ex	400K	clinical documents	A	using annotated logical forms on i2b2 dataset
HotpotQA (Yang et al. 2018)	Ex (+YN)	113K	Wikipedia	C	multi-hop reasoning
ReCoRD (Zhang et al. 2018a)	Ex	120K	news article	C	commonsense reasoning, cloze query
Natural Questions (Kwiatkowski et al. 2019)	Ex (+YN)	323K	Wikipedia	Q/C	short/long answer styles
Quoref (Dasigi et al. 2019)	Ex	24K	Wikipedia	C	coreference resolution
TechQA (Castelli et al. 2019)	Ex (+NA)	1.4K	IT support	X	technical support domain, domain-adaptation

Table 4: Free-form MRC datasets.

Name	Ans	Size	Domain	Src	Feature
bAbI (Weston et al. 2015)	FF	10K * 20	generated text	A	prerequisite toy tasks
LAMBADA (Paperno et al. 2016)	FF	10K	narrative	C	hard language modeling
WikiReading (Hewlett et al. 2016)	FF	18M	Wikipedia	A	super large-scale dataset
MS MARCO (Bajaj et al. 2016)	FF	100K	web snippet	Q	description on web snippets
NarrativeQA (Kočiskỳ et al. 2018)	FF	45K	movie script	C	summary/full story tasks
DuReader (He et al. 2018)	FF	200K	web snippet	Q/C	Chinese, Baidu Search/Knows
QuAC (Choi et al. 2018)	FF (+YN)	100K	Wikipedia	C	dialogue-based, 14k dialogs
ShARC (Saeidi et al. 2018)	YN*	32K	web snippet	C	reasoning on rules taken from government documents
CoQA (Reddy, Chen, and Manning 2019)	FF (+YN)	127K	Wikipedia	C	dialogue-based, 8k dialogs
BoolQ (Clark et al. 2019a)	YN	16K	Wikipedia	Q/C	boolean questions, subset of Natural Questions
PubMedQA (Jin et al. 2019b)	YN	273.5K	PubMed	X/A	biomedical domain, 1k expert questions
DROP (Dua et al. 2019)	FF	96K	Wikipedia	C	discrete reasoning

7. Conclusion

This work comprehensively reviews the studies of MRC in the scopes of background, definition, development, influence, datasets, technical and benchmark highlights, trends, and opportunities. We first briefly introduced the history of MRC and the background of contextualized language models. Then, we discussed the role of contextualized language models and the influence of MRC to the NLP community. The previous technical advances were summarized in the framework of Encoder to Decoder. After going through the mechanisms of MRC systems, we showed the highlights in different stages of MRC studies. Finally, we summarized the trends and opportunities. The basic views we have arrived at are that 1) MRC boosts the progress from language processing to understanding; 2) the rapid improvement of MRC systems greatly benefits from the progress of CLMs; 3) the theme of MRC is gradually moving from shallow text matching to cognitive reasoning.

Appendix A: Machine Reading Comprehension Datasets

This appendix lists existing machine reading comprehension datasets along with their answer styles, dataset size, type of corpus, sourcing methods, and focuses. Part of the statistics is borrowed from Sugawara, Stenetorp, and Aizawa (2020). *Ans* denotes answer styles where *Ex* is answer extraction by selecting a span in the given context, and *FF* is free-form answering. *NA* denotes that unanswerable questions are involved, and YN means yes or no answers. *Size* indicates the size of the whole dataset, including training, development, and test sets. *Src* represents how the questions are sourced where *X* means questions written by experts, *C* by crowdworkers, *A* by machines with an automated manner, and *Q* are search-engine queries. Note that the boundary between cloze-style and multi-choice datasets is not clear sometimes; for example, some candidate choices may be provided for cloze tests, such as Story Cloze Test (Mostafazadeh et al. 2017) and CLOTH (Xie et al. 2018). In our taxonomy, we regard the fix-choice tasks whose candidates are in a fixed number as multi-choice. In addition, some datasets are composed of different types of subtasks; we classify them according to the main types with special notations in *Ans* column.

References

Alberti, Chris, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic qa corpora generation with roundtrip consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173.

Ando, Rie Kubota and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853.

Back, Seohyun, Sai Chetan Chinthakindi, Akhil Kedia, Haejun Lee, and Jaegul Choo. 2020. NeurQuRI: Neural question requirement inspector for answerability prediction in machine reading comprehension. In *International Conference on Learning Representations*.

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

Bajaj, Payal, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

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

Bao, Hangbo, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Lei Cui, Songhao Piao, and Ming Zhou. 2019. Inspecting unification of encoding and matching with

transformer: A case study of machine reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 14–18.

Blitzer, John, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 120–128.

Brown, Peter F, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Castelli, Vittorio, Rishav Chakravarti, Saswati Dana, Anthony Ferritto, Radu Florian, Martin Franz, Dinesh Garg, Dinesh Khandelwal, Scott McCarley, Mike McCawley, et al. 2019. The techqa dataset. *arXiv preprint arXiv:1911.02984*.

Charniak, Eugene, Yasemin Altun, Rodrigo de Salvo Braz, Benjamin Garrett, Margaret Kosmala, Tomer Moscovich, Lixin Pang, Changhee Pyo, Ye Sun, Wei Wy, et al. 2000. Reading comprehension programs in a statistical-language-processing class. In *Proceedings of the 2000 ANLP/NAACL Workshop on Reading comprehension tests as evaluation for computer-based language understanding sytems-Volume 6*, pages 1–5, Association for Computational Linguistics.

Chen, Danqi. 2018. *Neural Reading Comprehension and Beyond*. Ph.D. thesis, Stanford University.

Chen, Danqi, Jason Bolton, and Christopher D Manning. 2016. 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)*, pages 2358–2367.

Chen, Danqi, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. 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.

Chen, Yu, Lingfei Wu, and Mohammed J Zaki. 2019. Graphflow: Exploiting conversation flow with graph neural networks for conversational machine comprehension. *arXiv preprint arXiv:1908.00059*.

Chen, Zhipeng, Yiming Cui, Wentao Ma, Shijin Wang, and Guoping Hu. 2019. Convolutional spatial attention model for reading comprehension with multiple-choice questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6276–6283.

Cho, Kyunghyun, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

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

Clark, Christopher and Matt Gardner. 2018. 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.

Clark, Christopher, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019a. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Association for Computational Linguistics, Minneapolis, Minnesota.

Clark, Kevin, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019b. What does bert look at? an analysis of bertâĂŹs attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.

Clark, Kevin, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019c. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Clark, Peter, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *CoRR*, abs/1803.05457.

Cui, Yiming, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2019. Cross-lingual machine reading

comprehension. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1586–1595.

Cui, Yiming, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. 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.

Cullingford, Richard E. 1977. Controlling inference in story understanding. In *IJCAI*, volume 77, page 17, Citeseer.

Dai, Zihang, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988.

Dalvi, Bhavana, Lifu Huang, Niket Tandon, Wen-tau Yih, and Peter Clark. 2018. Tracking state changes in procedural text: a challenge dataset and models for process paragraph 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)*, pages 1595–1604, Association for Computational Linguistics.

Dasigi, Pradeep, Nelson F. Liu, Ana Marasovic, Noah A. Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5927–5934, Association for Computational Linguistics, Hong Kong, China.

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

Dhingra, Bhuwan, Hanxiao Liu, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. 2017. 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.

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

Ding, Ming, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. 2019. Cognitive graph for multi-hop reading comprehension at scale. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2694–2703.

Dong, Li, Furu Wei, Ming Zhou, and Ke Xu. 2015. Question answering over freebase with multi-column convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 260–269.

Dong, Li, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13042–13054.

Du, Xinya and Claire Cardie. 2017. Identifying where to focus in reading comprehension for neural question generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2067–2073.

Du, Xinya, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352.

Dua, Dheeru, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378.

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

Ettinger, Allyson. 2020. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for*

Computational Linguistics, 8:34–48.
Evans, Jonathan St BT. 1984. Heuristic and analytic processes in reasoning. *British Journal of Psychology*, 75(4):451–468.
Evans, Jonathan St BT. 2003. In two minds: dual-process accounts of reasoning. *Trends in cognitive sciences*, 7(10):454–459.
Evans, Jonathan St BT. 2017. Dual process theory: perspectives and problems. In *Dual process theory 2.0*. Routledge, pages 137–155.
Fisch, Adam, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. Mrqa 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13.
Gao, Shuyang, Sanchit Agarwal, Tagyoung Chung, Di Jin, and Dilek Hakkani-Tur. 2020. From machine reading comprehension to dialogue state tracking: Bridging the gap. *arXiv preprint arXiv:2004.05827*.
Gao, Shuyang, Abhishek Sethi, Sanchit Agarwal, Tagyoung Chung, and Dilek Hakkani-Tur. 2019a. Dialog state tracking: A neural reading comprehension approach. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 264–273.
Gao, Yifan, Lidong Bing, Piji Li, Irwin King, and Michael R Lyu. 2019b. Generating distractors for reading comprehension questions from real examinations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6423–6430.
Gardner, Matt, Jonathan Berant, Hannaneh Hajishirzi, Alon Talmor, and Sewon Min. 2019. Question answering is a format; when is it useful? *arXiv preprint arXiv:1909.11291*.
Goldberg, Yoav. 2019. Assessing bert's syntactic abilities. *arXiv preprint arXiv:1901.05287*.
Goyal, Yash, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
Gupta, Nitish, Kevin Lin, Dan Roth, Sameer Singh, and Matt Gardner. 2019. Neural module networks for reasoning over text. *arXiv preprint arXiv:1912.04971*.
Guu, Kelvin, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*.
Habernal, Ivan, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1930–1940.
He, Wei, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, et al. 2018. 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.
Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pages 1693–1701.
Hewlett, Daniel, Alexandre Lacoste, Llion Jones, Illia Polosukhin, Andrew Fandrianto, Jay Han, Matthew Kelcey, and David Berthelot. 2016. WikiReading: A novel large-scale language understanding task over wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1545, Association for Computational Linguistics, Berlin, Germany.
Hill, Felix, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.
Hirschman, Lynette, Marc Light, Eric Breck, and John D Burger. 1999. Deep read: A reading comprehension system. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 325–332, Association for Computational Linguistics.
Hochreiter, Sepp and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
Htut, Phu Mon, Samuel Bowman, and Kyunghyun Cho. 2018. Training a ranking function for open-domain question answering. In *Proceedings of the 2018 Conference of the North American Chapter of*

the Association for Computational Linguistics: Student Research Workshop, pages 120–127.

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

Hu, Minghao, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019b. Retrieve, read, rerank: Towards end-to-end multi-document reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2285–2295.

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

Hu, Minghao, Yuxing Peng, and Xipeng Qiu. 2017. Mnemonic reader for machine comprehension. *CoRR*, abs/1705.02798.

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

Huang, Hsin-Yuan, Eunsol Choi, and Wen tau Yih. 2019. FlowQA: Grasping flow in history for conversational machine comprehension. In *International Conference on Learning Representations*.

Huang, Lifu, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. *arXiv preprint arXiv:1909.00277*.

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

Jiang, Yichen, Nitish Joshi, Yen-Chun Chen, and Mohit Bansal. 2019. Explore, propose, and assemble: An interpretable model for multi-hop reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2714–2725.

Jiao, Xiaoqi, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*.

Jin, Di, Shuyang Gao, Jiun-Yu Kao, Tagyoung Chung, and Dilek Hakkani-tur. 2019a. Mmm: Multi-stage multi-task learning for multi-choice reading comprehension. *arXiv preprint arXiv:1910.00458*.

Jin, Qiao, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019b. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Association for Computational Linguistics, Hong Kong, China.

Joshi, Mandar, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

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

Kadlec, Rudolf, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. 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)*, pages 908–918.

Kahneman, Daniel. 2011. *Thinking, fast and slow*. Macmillan.

Kembhavi, Aniruddha, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *the IEEE Conference on Computer Vision and Pattern Recognition*.

Keskar, Nitish Shirish, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Unifying question answering, text classification, and regression via span extraction. *arXiv preprint arXiv:1904.09286*.

Khashabi, Daniel, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 252–262, Association for Computational Linguistics.

Kim, Yanghoon, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2019. Improving neural question generation using answer separation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6602–6609.

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

Kwiatkowski, Tom, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Labutov, Igor, Bishan Yang, Anusha Prakash, and Amos Azaria. 2018. Multi-relational question answering from narratives: Machine reading and reasoning in simulated worlds. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 833–844, Association for Computational Linguistics.

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

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

Lee, Kyungjae, Sunghyun Park, Hojae Han, Jinyoung Yeo, Seung-won Hwang, and Juho Lee. 2019. Learning with limited data for multilingual reading comprehension. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2833–2843.

Lehnert, Wendy G. 1977. A conceptual theory of question answering. In *Proceedings of the 5th international joint conference on Artificial intelligence-Volume 1*, pages 158–164.

Li, Junlong, Zhuosheng Zhang, and Hai Zhao. 2020. Multi-choice dialogue-based reading comprehension with knowledge and key turns. *arXiv preprint arXiv:2004.13988*.

Li, Xiaoya, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2019a. A unified mrc framework for named entity recognition. *arXiv preprint arXiv:1910.11476*.

Li, Xiaoya, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019b. Entity-relation extraction as multi-turn question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1340–1350.

Li, Zuchao, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. 2019c. Explicit sentence compression for neural machine translation. *arXiv preprint arXiv:1912.11980*.

Lin, Bill Yuchen, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2822–2832.

Lin, Chin-Yew. 2004. Looking for a few good metrics: Automatic summarization evaluation-how many samples are enough? In *NTCIR*.

Liu, Pengyuan, Yuning Deng, Chenghao Zhu, and Han Hu. 2019a. Xcmrc: Evaluating cross-lingual machine reading comprehension. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 552–564, Springer.

Liu, Shanshan, Xin Zhang, Sheng Zhang, Hui Wang, and Weiming Zhang. 2019b. Neural machine reading comprehension: Methods and trends. *Applied Sciences*, 9(18):3698.

Liu, Xiaodong, Yelong Shen, Kevin Duh, and Jianfeng Gao. 2018. Stochastic answer networks for machine reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1694–1704.

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

Lu, Jiasen, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.

Ma, Kaixin, Jonathan Francis, Quanyang Lu, Eric Nyberg, and Alessandro Oltramari. 2019. Towards generalizable neuro-symbolic systems for commonsense question answering. *arXiv preprint arXiv:1910.14087*.

Ma, Kaixin, Tomasz Jurczyk, and Jinho D Choi. 2018. 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)*, pages 2039–2048.

McCann, Bryan, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Mihaylov, Todor, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Association for Computational Linguistics, Brussels, Belgium.

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

Min, Sewon, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and robust question answering from minimal context over documents. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1725–1735.

Mostafazadeh, Nasrin, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849.

Mostafazadeh, Nasrin, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. LSDSem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51, Association for Computational Linguistics.

Nishida, Kyosuke, Itsumi Saito, Kosuke Nishida, Kazutoshi Shinoda, Atsushi Otsuka, Hisako Asano, and Junji Tomita. 2019. Multi-style generative reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2273–2284.

Onishi, Takeshi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. 2016. 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, Association for Computational Linguistics.

Ostermann, Simon, Ashutosh Modi, Michael Roth, Stefan Thater, and Manfred Pinkal. 2018. 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)*, European Language Resources Association (ELRA).

Ostermann, Simon, Michael Roth, and Manfred Pinkal. 2019. MCScript2.0: A machine comprehension corpus focused on script events and participants. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 103–117, Association for Computational Linguistics, Minneapolis, Minnesota.

Pampari, Anusri, Preethi Raghavan, Jennifer Liang, and Jian Peng. 2018. emrqa: A large corpus for question answering on electronic medical records. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2357–2368.

Pan, Boyuan, Hao Li, Zhou Zhao, Bin Cao, Deng Cai, and Xiaofei He. 2017. MEMEN: Multi-layer embedding with memory networks for machine comprehension. *arXiv preprint arXiv:1707.09098*.

Pan, Lin, Rishav Chakravarti, Anthony Ferritto, Michael Glass, Alfio Gliozzo, Salim Roukos, Radu Florian, and Avirup Sil. 2019a. Frustratingly easy natural question answering. *arXiv preprint arXiv:1909.05286*.

Pan, Xiaoman, Kai Sun, Dian Yu3 Jianshu Chen, Heng Ji, Claire Cardie, and Dong Yu. 2019b. Improving question answering with external knowledge. In *EMNLP 2019 MRQA Workshop*, page 27.

Paperno, Denis, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. 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)*, pages 1525–1534, Association for Computational Linguistics, Berlin, Germany.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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.

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

Peters, Matthew E, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Qiao, Yifan, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the behaviors of bert in ranking. *arXiv preprint arXiv:1904.07531*.

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

Qiu, Lin, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019b. Dynamically fused graph network for multi-hop reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6140–6150.

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

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Rajpurkar, Pranav, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.

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

Ran, Qiu, Peng Li, Weiwei Hu, and Jie Zhou. 2019a. Option comparison network for multiple-choice reading comprehension. *arXiv preprint arXiv:1903.03033*.

Ran, Qiu, Yankai Lin, Peng Li, Jie Zhou, and Zhiyuan Liu. 2019b. Numnet: Machine reading comprehension with numerical reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2474–2484.

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

Richardson, Matthew, Christopher JC Burges, and Erin Renshaw. 2013. 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.

Riloff, Ellen and Michael Thelen. 2000. A rule-based question answering system for reading comprehension tests. In

Proceedings of the 2000 ANLP/NAACL Workshop on Reading comprehension tests as evaluation for computer-based language understanding sytems-Volume 6, pages 13–19, Association for Computational Linguistics.

Rogers, Anna, Olga Kovaleva, Matthew Downey, and Anna Rumshisky. 2020. Getting closer to AI complete question answering: A set of prerequisite real tasks. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Saeidi, Marzieh, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2087–2097.

Saha, Amrita, Rahul Aralikatte, Mitesh M. Khapra, and Karthik Sankaranarayanan. 2018. 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)*, pages 1683–1693, Association for Computational Linguistics.

Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Sap, Maarten, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035.

Schlegel, Viktor, Marco Valentino, André Freitas, Goran Nenadic, and Riza Batista-Navarro. 2020. A framework for evaluation of machine reading comprehension gold standards. *arXiv preprint arXiv:2003.04642*.

Seo, Minjoon, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *ICLR 2017*.

Shao, Nan, Yiming Cui, Ting Liu, Shijin Wang, and Guoping Hu. 2020. Is graph structure necessary for multi-hop reasoning? *arXiv preprint arXiv:2004.03096*.

Shen, Sheng, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. 2019. Q-bert: Hessian based ultra low precision quantization of bert. *arXiv preprint arXiv:1909.05840*.

Shen, Yelong, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. 2017. 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.

Shoeybi, Mohammad, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

Song, Kaitao, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

Song, Linfeng, Zhiguo Wang, Mo Yu, Yue Zhang, Radu Florian, and Daniel Gildea. 2018. Exploring graph-structured passage representation for multi-hop reading comprehension with graph neural networks. *arXiv preprint arXiv:1809.02040*.

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

Speer, Robyn, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Sugawara, Saku, Kentaro Inui, Satoshi Sekine, and Akiko Aizawa. 2018. What makes reading comprehension questions easier? In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4208–4219.

Sugawara, Saku, Pontus Stenetorp, and Akiko Aizawa. 2020. Prerequisites for explainable machine reading comprehension: A position paper. *arXiv preprint arXiv:2004.01912*.

Sugawara, Saku, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2019. Assessing the benchmarking capacity of machine reading comprehension datasets. *arXiv preprint arXiv:1911.09241*.

Sun, Kai, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019a. Dream: A challenge data set and models for dialogue-based reading comprehension. *Transactions of the Association for Computational Linguistics*,

7:217–231.
Sun, Kai, Dian Yu, Dong Yu, and Claire Cardie. 2019b. Improving machine reading comprehension with general reading strategies. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2633–2643.
Suster, Simon and Walter Daelemans. 2018. 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)*, pages 1551–1563, Association for Computational Linguistics.
Sutcliffe, Richard, Anselmo Peñas, Eduard Hovy, Pamela Forner, Álvaro Rodrigo, Corina Forascu, Yassine Benajiba, and Petya Osenova. 2013. Overview of QA4MRE main task at CLEF 2013. *Working Notes, CLEF*.
Sutskever, Ilya, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.
Talmor, Alon and Jonathan Berant. 2019. Multiqa: An empirical investigation of generalization and transfer in reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4911–4921.
Talmor, Alon, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Association for Computational Linguistics, Minneapolis, Minnesota.
Tan, Chuanqi, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2018. S-net: From answer extraction to answer synthesis for machine reading comprehension. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
Tandon, Niket, Gerard De Melo, and Gerhard Weikum. 2017. Webchild 2.0: Fine-grained commonsense knowledge distillation. In *Proceedings of ACL 2017, System Demonstrations*, pages 115–120.
Tang, Min, Jiaran Cai, and Hankz Hankui Zhuo. 2019. Multi-matching network for multiple choice reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7088–7095.
Tay, Yi, Luu Anh Tuan, and Siu Cheung Hui. 2018. Multi-range reasoning for machine comprehension. *arXiv preprint arXiv:1803.09074*.
Trischler, Adam, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. Newsqa: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200.
Tu, Ming, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Graph sequential network for reasoning over sequences. *arXiv preprint arXiv:2004.02001*.
Tu, Ming, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2019. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. *arXiv preprint arXiv:1911.00484*.
Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.
Vinyals, Oriol, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in neural information processing systems*, pages 2692–2700.
Wallace, Eric, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. 2019. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*, 7:387–401.
Wang, Huazheng, Zhe Gan, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, and Hongning Wang. 2019. Adversarial domain adaptation for machine reading comprehension. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2510–2520.
Wang, Shuohang and Jing Jiang. 2016. Machine comprehension using Match-LSTM and answer pointer. *arXiv preprint arXiv:1608.07905*.

Wang, Shuohang, Mo Yu, Jing Jiang, and Shiyu Chang. 2018a. A co-matching model for multi-choice reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 746–751.

Wang, Shuohang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2018b. Evidence aggregation for answer re-ranking in open-domain question answering. In *International Conference on Learning Representations*.

Wang, Wei, Bin Bi, Ming Yan, Chen Wu, Jiangnan Xia, Zuyi Bao, Liwei Peng, and Luo Si. 2020a. Structbert: Incorporating language structures into pre-training for deep language understanding. In *International Conference on Learning Representations*.

Wang, Wenhui, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020b. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *arXiv preprint arXiv:2002.10957*.

Wang, Wenhui, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. 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.

Wang, Yizhong, Kai Liu, Jing Liu, Wei He, Yajuan Lyu, Hua Wu, Sujian Li, and Haifeng Wang. 2018c. Multi-passage machine reading comprehension with cross-passage answer verification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1918–1927.

Wang, Zhen, Jiachen Liu, Xinyan Xiao, Yajuan Lyu, and Tian Wu. 2018d. Joint training of candidate extraction and answer selection for reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1724.

Wason, Peter C and J St BT Evans. 1974. Dual processes in reasoning? *Cognition*, 3(2):141–154.

Welbl, Johannes, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302.

Weston, Jason, Antoine Bordes, Sumit Chopra, and Tomas Mikolov. 2015. Towards AI-complete question answering: a set of prerequisite toy tasks. In *International Conference on Learning Representations*.

Xie, Qizhe, Guokun Lai, Zihang Dai, and Eduard Hovy. 2018. Large-scale cloze test dataset created by teachers. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2344–2356.

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

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

Xiong, Caiming, Victor Zhong, and Richard Socher. 2018. DCN+: Mixed objective and deep residual coattention for question answering. In *International Conference on Learning Representations*.

Xu, Canwen, Jiaxin Pei, Hongtao Wu, Yiyu Liu, and Chenliang Li. 2020. Matinf: A jointly labeled large-scale dataset for classification, question answering and summarization. *arXiv preprint arXiv:2004.12302*.

Xu, Yichong, Xiaodong Liu, Yelong Shen, Jingjing Liu, and Jianfeng Gao. 2018. Multi-task learning with sample re-weighting for machine reading comprehension. *arXiv preprint arXiv:1809.06963*.

Yagcioglu, Semih, Aykut Erdem, Erkut Erdem, and Nazli Ikizler-Cinbis. 2018. Recipeqa: A challenge dataset for multimodal comprehension of cooking recipes. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1358–1368.

Yan, Ming, Jiangnan Xia, Chen Wu, Bin Bi, Zhongzhou Zhao, Ji Zhang, Luo Si, Rui Wang, Wei Wang, and Haiqing Chen. 2019. A deep cascade model for multi-document reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7354–7361.

Yang, Wei, Yuqing Xie, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019a. Data augmentation for bert fine-tuning in open-domain question answering. *arXiv preprint arXiv:1904.06652*.

Yang, Wei, Yuqing Xie, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019b.

Data augmentation for bert fine-tuning in open-domain question answering. *arXiv preprint arXiv:1904.06652*.

Yang, Zhilin, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019c. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Yang, Zhilin, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W. Cohen, and Ruslan Salakhutdinov. 2017a. Words or characters? fine-grained gating for reading comprehension. In *Proceedings of the International Conference on Learning Representations (ICLR 2017)*.

Yang, Zhilin, Junjie Hu, Ruslan Salakhutdinov, and William Cohen. 2017b. Semi-supervised qa with generative domain-adaptive nets. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1040–1050.

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

Yu, Adams Wei, David Dohan, Quoc Le, Thang Luong, Rui Zhao, and Kai Chen. 2018. QANet: Combining local convolution with global self-attention for reading comprehension. In *International Conference on Learning Representations*.

Yu, Adams Wei, Hongrae Lee, and Quoc Le. 2017. Learning to skim text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1880–1890.

Zellers, Rowan, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Association for Computational Linguistics.

Zellers, Rowan, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Association for Computational Linguistics, Florence, Italy.

Zhang, Sheng, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018a. Record: Bridging the gap between human and machine commonsense reading comprehension. *arXiv preprint arXiv:1810.12885*.

Zhang, Shuailiang, Hai Zhao, Yuwei Wu, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2019a. Dual co-matching network for multi-choice reading comprehension. *arXiv preprint arXiv:1901.09381*.

Zhang, Shuailiang, Hai Zhao, Yuwei Wu, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2020a. DCMN+: Dual co-matching network for multi-choice reading comprehension. In *AAAI*.

Zhang, Shuailiang, Hai Zhao, and Junru Zhou. 2020. Semantics-aware inferential network for natural language understanding. *arXiv preprint arXiv:2004.13338*.

Zhang, Xin, An Yang, Sujian Li, and Yizhong Wang. 2019b. Machine reading comprehension: a literature review. *arXiv preprint arXiv:1907.01686*.

Zhang, Yiqing, Hai Zhao, and Zhuosheng Zhang. 2019. Examination-style reading comprehension with neural augmented retrieval. In *2019 International Conference on Asian Language Processing (IALP)*, pages 182–187, IEEE.

Zhang, Zhuosheng, Kehai Chen, Rui Wang, Masao Utiyama, Eiichiro Sumita, Zuchao Li, and Hai Zhao. 2019c. Neural machine translation with universal visual representation. In *International Conference on Learning Representations (ICLR)*.

Zhang, Zhuosheng, Yafang Huang, and Hai Zhao. 2018. Subword-augmented embedding for cloze reading comprehension. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 1802–âĂŞ1814.

Zhang, Zhuosheng, Yafang Huang, and Hai Zhao. 2019. Open vocabulary learning for neural chinese pinyin ime. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1584–1594.

Zhang, Zhuosheng, Yafang Huang, Pengfei Zhu, and Hai Zhao. 2018b. Effective character-augmented word embedding for machine reading comprehension. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 27–39, Springer.

Zhang, Zhuosheng, Jiangtong Li, Pengfei Zhu, and Hai Zhao. 2018c. Modeling

multi-turn conversation with deep utterance aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 3740–3752.

Zhang, Zhuosheng, Yuwei Wu, Zuchao Li, and Hai Zhao. 2019d. Explicit contextual semantics for text comprehension. In *Proceedings of the 33rd Pacific Asia Conference on Language, Information and Computation (PACLIC 33)*.

Zhang, Zhuosheng, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020b. Semantics-aware bert for language understanding. In *the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-2020)*.

Zhang, Zhuosheng, Yuwei Wu, Junru Zhou, Sufeng Duan, Hai Zhao, and Rui Wang. 2020c. SG-Net: Syntax-guided machine reading comprehension. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*.

Zhang, Zhuosheng, Junjie Yang, and Hai Zhao. 2020. Retrospective reader for machine reading comprehension. *arXiv preprint arXiv:2001.09694*.

Zhang, Zhuosheng and Hai Zhao. 2018. One-shot learning for question-answering in gaokao history challenge. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 449–461.

Zhang, Zhuosheng, Hai Zhao, Kangwei Ling, Jiangtong Li, Shexia He, and Guohong Fu. 2019e. Effective subword segmentation for text comprehension. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, 27(11):1664–1674.

Zhou, Junru, Zhuosheng Zhang, and Hai Zhao. 2019. LIMIT-BERT: Linguistic informed multi-task bert. *arXiv preprint arXiv:1910.14296*.

Zhu, Haichao, Li Dong, Furu Wei, Wenhui Wang, Bing Qin, and Ting Liu. 2019. Learning to ask unanswerable questions for machine reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4238–4248.

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

Zhu, Pengfei, Zhuosheng Zhang, Jiangtong Li, Yafang Huang, and Hai Zhao. 2018b. Lingke: a fine-grained multi-turn chatbot for customer service. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 108–112.

Zhu, Pengfei, Hai Zhao, and Xiaoguang Li. 2020. Dual multi-head co-attention for multi-choice reading comprehension. *arXiv preprint arXiv:2001.09415*.