Label Distribution Augmented Maximum Likelihood Estimation for Reading Comprehension

Lixin Su, Jiafeng Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng
CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
sulixinict@gmail.com,{guojiafeng,fanyixing,lanyanyan,cxq}@ict.ac.cn

ABSTRACT

Reading comprehension (RC) aims to locate a text span from a context passage to answer the given question. Despite the effectiveness of modern neural RC models, most existing work relies on maximum likelihood estimation (MLE) and ignores the structure of the output space. That is during training, one treats all the text spans do not match the ground truth as equally poor, leading to overconfident predictions on ground truth labels and reduced generalization ability in test. One way to bridge the gap between training and test is to take into account the task reward of alternative outputs using the reinforcement learning (RL) algorithms, which is often deficient in optimization as compared with MLE. In this paper, we propose a new learning criterion for the RC task which combines the merits of both MLE and RL-based methods. Specifically, we show that we are able to derive the distribution of the outputs, i.e., label distribution, using their corresponding task rewards based on the decomposition property of the RC problem. We then optimize the RC model by directly learning towards the auxiliary label distribution, instead of the ground truth label, using the MLE framework. In this way, we can make use of the structure of the output space for better generalization (as RL) via efficient optimization (as MLE). We name our approach as Label Distribution augmented MLE (LD-MLE), which is a general learning criterion that could be adopted by almost all the existing RC models. Experiments on three representative benchmark datasets demonstrate that RC models learned with the LD-MLE criterion can achieve consistently improved results over those based on the traditional MLE and RL-based criteria.

CCS CONCEPTS

• **Information systems → Question answering**.

KEYWORDS

reading comprehension, question answering, label smoothing

WSDM '20, February 3–7, 2020, Houston, TX, USA

ACM ISBN 978-1-4503-6822-3/20/02…$15.00
https://doi.org/10.1145/3336191.3371835

ACM Reference Format:
Lixin Su, Jiafeng Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng. 2020. Label Distribution Augmented Maximum Likelihood Estimation for Reading Comprehension. In *The Thirteenth ACM International Conference on Web Search and Data Mining (WSDM '20), February 3–7, 2020, Houston, TX, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3336191.3371835

1 INTRODUCTION

Reading comprehension (RC), aiming to understand natural texts to answer questions, is a challenging task in natural language processing [10, 13, 16, 27, 28] and has became an important component in QA system [1]. Without loss of generality, the major task of RC can be defined as an answer span prediction problem, i.e., to predict the start and end positions of an answer span in a context passage given a question. With the development of deep learning techniques, state-of-the-art performances have been achieved by modern neural RC models [4, 11, 20, 40, 42].

Despite the effectiveness of neural RC models, most existing work relies on maximum likelihood estimation (MLE) for learning and ignores the structure of the output space. That is during training, RC models focus on maximizing the likelihood of the target answer span (i.e., the ground truth start and end labels). They treat all the text spans that do not match the ground truth as equally poor, regardless of their structural proximity to the ground truth. For example, as shown in Figure 1, given the context passage and the question, the ground truth answer is "18th and 19th centuries" labeled by the pair ($s = 135, e = 138$), where s and e denotes the start and end position of the answer span, respectively. Considering two alternative outputs, i.e., "the 18th and 19th" ($s = 134, e = 137$) and "no continuity" ($s = 143, e = 144$), apparently Span 1 is preferable to Span 2 since it overlaps with the ground truth span. Such preference is usually reflected in evaluation with metrics based on n-grams. However, these two outputs are equally punished during training under the MLE criterion, making it inconsistent with the test evaluation. As a result, RC models learned under the MLE criterion may become overconfident on the ground truth label [34, 42], leading to overfitting and reduced generalization ability on test instances.

One way to eliminate this discrepancy between training and test in RC is to take into account the alternative outputs beyond the ground truth and optimize the task reward (e.g., F1) that matters for test evaluation over them. However, since such task reward is usually not differentiable, reinforcement learning (RL) techniques have been adopted to maximize the expected reward [11, 42]. For example, DCN+[42] proposed to use self-critical policy learning [15, 33] to optimize the expected reward in RC. R.M-Reader [11] further

Context:... Johannes Wallmann argues that Luther's writings against the Jews were largely ignored in the 18th and 19th centuries, and that there was no continuity between Luther's thought ...
Question: When was Luther's writings about the Jews ignored, according to Johannes Wallmann ?
Answer span: 18th and 19th centuries (s=135,e=138)
Span 1: the 18th and 19th (s=134,e=137)
Span 2: no continuity (s=143, e=144)

Figure 1: An example from the SQuAD dataset. Span 1 and Span 2 are two alternative outputs beyond the ground truth answer span. s and e denotes the start and end position of the answer span in the context passage, respectively.

leveraged dynamic-critical reinforcement learning to address the convergence suppression problem occurred in DCN+. However, these RL-based methods face significant challenges in optimization [23]: gradients need to be estimated via sampling from the model output, which is a non-stationary distribution, leading to high variance in gradients and difficulty in convergence.

In this paper, we introduce a new learning criterion for the RC task which combines the merits of both MLE and RL-based methods, namely *label distribution augmented MLE* (*LD-MLE* for short). It is also a general learning criterion that could be adopted by almost all the existing RC models. Specifically, we show that the traditional MLE objective in RC can be viewed as optimizing the KL-divergence between the target distribution and the model's distribution, where the target distribution is a Dirac distribution on the ground truth answer span. To make use of the structure of the output space, we aim to take into account the task reward of alternative outputs beyond the ground truth as RL-based methods. Different from the RL-based methods which relies on sampling, we show that we can derive a new target (label) distribution that summarizes the task rewards of all the possible outputs based on the decomposition property of the RC problem. In this way, we can replace the original Dirac distribution with this new label distribution in MLE for direct optimization. As we can see, our LD-MLE takes into account the structure of the output space within MLE framework, thus combines the advantages of RL-based methods with the computational efficiency and simplicity of MLE.

We conduct empirical experiments to verify the effectiveness of our proposed LD-MLE criterion. We take into account a variety of representative neural RC models, including BiDAF [30], SAN [20] and BERT [4]. We also use several widely adopted benchmark datasets, including the SQuAD dataset [28] which represents a basic RC task, and the MS MARCO [22] and CoQA [29] datasets which represent two popular variants of the RC task. We show that neural RC models learned with the LD-MLE criterion can achieve consistently improved results over those based on the traditional MLE and RL-based criteria on all these benchmark datasets. We also conduct extensive analysis to gain a better understanding of the advantage of the LD-MLE criterion.

The remainder of this paper is organized as follows. In Section 2, we introduce some backgrounds of our work. We then describe our LD-MLE method in detail and discuss its connection and difference with several existing techniques in Section 3. The experimental results are reported in Section 4. We review the related work in Section 5, and conclude the whole paper in Section 6.

2 BACKGROUND

Before diving into our method, we first introduce some background work in this section, including the formulation of the RC task and the MLE learning criterion.

2.1 Reading Comprehension

Reading comprehension has embraced a booming in recent NLP research thanks to a variety of publicly available benchmark collections [1], e.g., CNN/DAILY [10], SQuAD [28], and NewsQA [35]. Although there have been different variants of the RC task [22, 29, 44], a basic definition of RC is to locate the answer for a given question from a context passage. Formally, the context passage $C = \{c_1, c_2, \ldots c_m\}$ consists of a sequence of m words and the question $Q = \{q_1, q_2, \ldots, q_n\}$ consists of a sequence of n words. The answer $A = \{c_s, \ldots, c_e\}$ is assumed to be a contiguous text span in the context passage, where s and e denotes the corresponding start and end position, respectively. In the following, we will use $a = (s, e)$ to denote the answer span position and $a^* = (s^*, e^*)$ to denote the ground truth.

Typically, an RC model f takes Q and C as inputs and produces two vectors $\mathbf{s} \in \mathbb{R}^m$ and $\mathbf{e} \in \mathbb{R}^m$,

$$\mathbf{s}, \mathbf{e} = f(Q, C), \tag{1}$$

where $\mathbf{s}_i$ and $\mathbf{e}_i$ denotes the possibility of position i in the context passage to be the start and end term of an answer span, respectively.

The predicted answer position is then obtained by

$$\hat{s}, \hat{e} = \text{argmax}_{i,j} \mathbf{s}_i \cdot \mathbf{e}_j, \text{subject to } i < j,$$

and the predicted answer text is $\hat{A} = \{c_{\hat{s}}, \ldots, c_{\hat{e}}\}$.

Recently, RC models with deep learning techniques have achieved a great success [29]. Without loss of generality, a neural RC model often consists of four stacked layers, namely the embedding layer, the encoding layer, the interaction layer, and the answer layer.

- The embedding layer converts each word in the context passage and the question into a real-valued vector to obtain the embedding matrix $\mathbf{E}^C \in \mathbb{R}^{m \times d}$ and $\mathbf{E}^Q \in \mathbb{R}^{n \times d}$, where d denotes the embedding dimension.
- The encoding layer attempts to encode the context information to enrich the representation of each word in the context passage and the question.
- The interaction layer makes the question and the context passage interact with each other through certain attention mechanism to obtain a question-aware passage representation $\mathbf{H}^C \in \mathbb{R}^{m \times h}$.
- The answer layer then predicts the vectors $\mathbf{s}$ and $\mathbf{e}$ based on the passage representation $\mathbf{H}^C$.

Most RC models follow this design paradigm but apply different structures in each layer, e.g., BiDAF [30], SAN [20] and FusionNet [12]. BERT [4] is a little bit different, but can be viewed as

merging the encoding layer and the interaction layer together via the Transformer structure.

2.2 The MLE Criterion

Most existing neural RC models rely on MLE for model learning. Specifically, the MLE criterion is to maximize the log-likelihood of the ground truth answers as follows:

$$\begin{aligned}\mathcal{L}_{MLE}(\theta) &= -\log p(a^*|Q,C;\theta) \\ &= -\log p_s(s^*|Q,C;\theta)p_e(e^*|s^*,Q,C;\theta)\end{aligned} \tag{2}$$

where

$$p_s(\cdot|Q,C;\theta) = \text{softmax}(\mathbf{s}), p_e(\cdot|Q,C;\theta) = \text{softmax}(\mathbf{e}),$$

p_s denotes the predicted probability of the start position , and p_e denotes the predicted probability of the end position. In most existing RC work, the coupled prediction problem $p_s(s)p_e(e|s)$ is usually turned into two de-coupled ones, i.e., $p_s(s)$ and $p_e(e)$, by implicitly modeling the dependence via the dependent computation of the vector $\mathbf{e}$ and the vector $\mathbf{s}$ [30]. In this way, the above MLE learning objective turns into the following simplified form,

$$\begin{aligned}\mathcal{L}_{MLE}(\theta) &= -\log p_s(s^*|Q,C;\theta)p_e(e^*|Q,C;\theta) \\ &= -\log p_s(s^*|Q,C;\theta) - \log p_e(e^*|Q,C;\theta).\end{aligned} \tag{3}$$

It is worth to note that the MLE objective can also be written in the form of the KL divergence,

$$\begin{aligned}\mathcal{L}_{MLE}(\theta) &= -\sum_s \delta_{s^*}(s)\log p_s(s|Q,C;\theta) \\ &\quad -\sum_e \delta_{e^*}(e)\log p_e(e|Q,C;\theta) \\ &= -\sum_s \delta_{s^*}(s)\log\frac{p_s(s|Q,C;\theta)}{\delta_{s^*}(s)} + \sum_s \delta_{s^*}(s)\log\delta_{s^*}(s) \\ &\quad -\sum_e \delta_{e^*}(e)\log\frac{p_e(e|Q,C;\theta)}{\delta_{e^*}(e)} + \sum_e \delta_{e^*}(e)\log\delta_{e^*}(e) \\ &= D_{KL}(\delta_{s^*}||p_s) + D_{KL}(\delta_{e^*}||p_e) + const,\end{aligned} \tag{4}$$

where δ_{s^*} and δ_{e^*} denotes the Dirac distribution of ground truth start and end position respectively, i.e.,$\delta_{s^*}(s^*) = 1$ else $\delta_{s^*}(s) = 0$ for other s. We can clearly see that MLE is to minimizing the distance between the target $\delta_{s^*}/\delta_{e^*}$ distribution and model's distribution p_s/p_e.

While there has been a lot of effort dedicated to designing new neural RC model structures, little has been made on the learning criterion of neural RC models, which is the focus of this paper.

3 OUR METHOD

In this section, we describe the Label Distribution augmented Maximum Likelihood Estimation (LD-MLE) in detail. We also provide some discussions to show the connections and differences of LD-MLE with some related techniques.

3.1 LD-MLE

As aforementioned, the MLE learning criterion ignores the structure of the output space by treating all the outputs that do not match the ground truth as equally poor, regardless of their structural proximity to the ground truth. This brings the discrepancy between training and test, leading to overfitting on the ground truth labels and reduced generalization ability.

In this work, we aim to take into account the alternative outputs beyond the ground truth for better model learning, meanwhile attempt to keep the optimization procedure simple and efficient. The key idea is that if we can derive a better target (label) distribution, which can convey the information of the output structure, we can then directly replace it into the MLE objective as shown in Equation (4) to achieve our purpose.

In the following, we try to derive the new target distribution. The overview of the derivation process is illustrated in Figure 2, which consists of the following three steps.

1. Define output distribution

Without loss of generality, given a ground truth answer a^* and a reward function r (e.g., the evaluation metric F1 as defined in Equation (8), we can calculate the reward for each possible output answer span a as $r(a, a^*)$. Note that we can assign large negative number to those illegal spans (i.e., $s > e$) to exclude them. Following the idea in [3, 5], we normalize these reward scores to obtain the distribution of the outputs as

$$q_a(a|a^*;\tau) = \frac{\exp(r(a,a^*)/\tau)}{\sum_a \exp(r(a,a^*)/\tau)}, \tag{5}$$

where τ is the hyper-parameter which controls the concentration of the distribution around a^*. Obviously, this distribution reflects how the task rewards distributed in the output space.

2. Decompose to label distribution

Based on the de-couple idea in RC as shown in Equation (3), we can obtain the following start/end label distribution by marginalizing Equation (5) with respect to all the possible end/start positions.

$$\begin{aligned}q_s(s|a^*;\tau) &= \sum_e q_a(s,e|a^*;\tau) \\ &= \frac{\sum_e \exp(r(s,e,a^*)/\tau)}{\sum_s\sum_e \exp(r(s,e,a^*)/\tau)},\end{aligned}$$

$$\begin{aligned}q_e(e|a^*;\tau) &= \sum_s q_a(s,e|a^*;\tau) \\ &= \frac{\sum_s \exp(r(s,e,a^*)/\tau)}{\sum_s\sum e \exp(r(s,e,a^*)/\tau)},\end{aligned}$$

The above marginalization step could be efficiently computed with the computational complexity of $O(m^2)$ given the ground truth label a^* and the evaluation metric r, where m denotes the length of the context passage which typically small in practice. Note that these two label distributions have summarized the reward information of all the possible outputs with respect to each term position in the context passage.

3. Integrate into MLE criterion

Now we replace the above two derived label distributions into Equation (4) and obtain our LD-MLE learning criterion as follows

$$\begin{aligned}\mathcal{L}_{LD-MLE}(\theta) &= -\sum_{s=1}^{m} q_s(s|a^*;\tau)\log p_s(s|Q,C;\theta) \\ &\quad -\sum_{e=1}^{m} q_e(e|a^*;\tau)\log p_e(e|Q,C;\theta).\end{aligned}$$

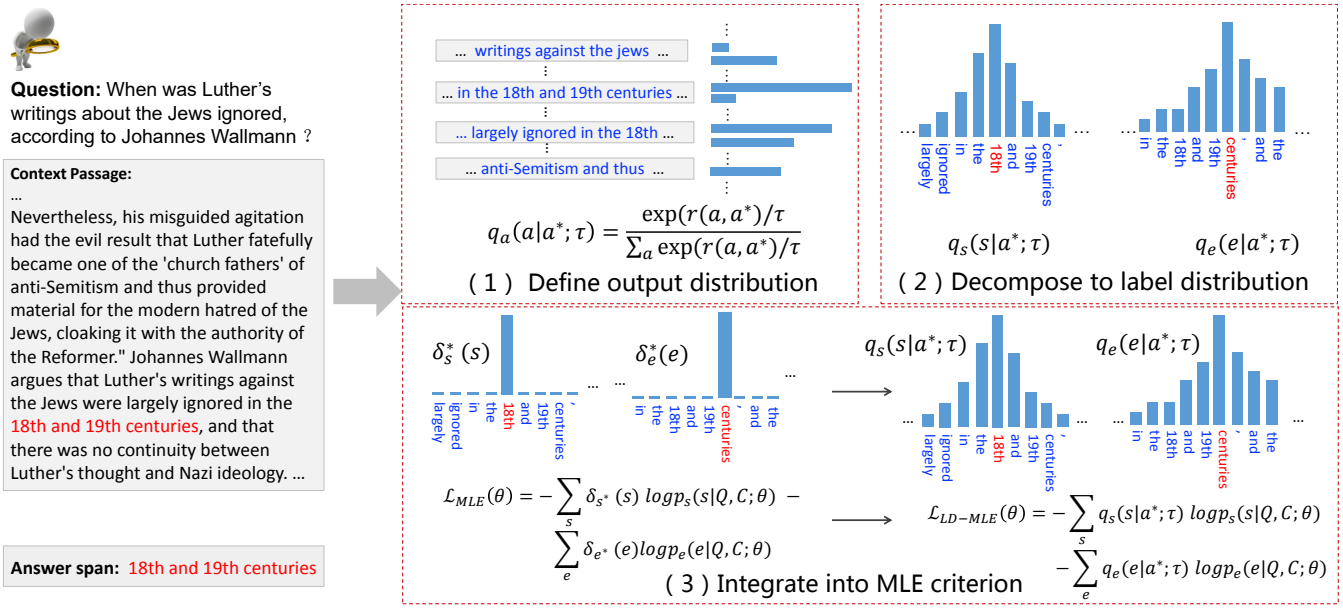


Figure 2: The derivation procedure of the LD-MLE learning criterion.

Now we can directly optimize this new label distribution augmented objective function for learning the RC model. It is not difficult to check that even though we replace the original (Dirac) distribution with the new augmented label distribution, the optimal solution of LD-MLE remains the same as the MLE criterion (i.e., a^*) which is an expected property of LD-MLE. Besides, we can see that this learning criterion is easy to implement in practice. It is also a general learning criterion that could be adopted by almost all the existing RC models.

3.2 Discussion

From the learning objective of LD-MLE, people might connect our method to the well-known Label Distribution Learning (LDL) approaches [6–8]. However, LDL mainly focuses on the scenario where one cannot obtain exact or complete labels to the task. For example, in facial age estimation, it is difficult to obtain exact and sufficient training data [8]. Therefore, some manually designed label distributions [6], e.g., normal distribution, are introduced to tackle that problem. Different from LDL, we introduce the label distribution to characterize the structure of the output space. The label distribution is derived based on the ground truth labels and the evaluation metrics.

Our method also shares some similar idea as label smoothing [34]. Label smoothing is a kind of regularization technique to encourage the model to be less confident on the ground truth labels to achieve better generalization ability. For example, in image classification, label smoothing takes part of the probability from the correct label to assign it to other labels equally [34]. Our LD-MLE method shares some similary to label smoothing in the sense that the derived label distribution could be viewed as the smoothing of the ground truth labels. However, our LD-MLE method smooths the target distribution with respect to the task reward, while traditional label smoothing usually does not take that into account.

Another line of highly related work to our method is the study on the reward augmented maximum likelihood (RAML) [3, 23]. However, existing work on RAML still requires sampling from the output space for optimization, which is apparently deficient. Such methods have only been applied on the machine translation task. To our best knowledge, our work is the first work to incorporate the task rewards over the output space into MLE for the RC problem.

Table 1: Statistics of different RC datasets.

Dataset	#Question	#Passage	#terms		
			P	Q	A
SQuAD	98,169	20,963	116.6	10.1	3.2
MS MARCO	808,731	8,069,749	56.4	6.4	9.2
CoQA	116,630	7,699	271.0	5.5	2.7

4 EXPERIMENTS

In this section, we conduct empirical experiments to verify the effectiveness of the LD-MLE criterion. Besides, we also provide in-depth analysis to gain a better understanding of the advantages of our method.

4.1 Experimental Settings

In this part, we describe the experimental settings, including the datasets, evaluation metrics, baseline methods, RC models, and their implementation details.

4.1.1 Datasets. We choose three representative RC datasets to conduct the experiments. Among these datasets, SQuAD is the

primary RC task, while MS MARCO and CoQA are two popular variants of RC tasks. The detailed descriptions of the three datasets are as follows:

- **SQuAD** [28] is a typical RC dataset that has been widely studied in academia [30, 38, 42]. Each context passage in SQuAD is a paragraph from Wikipedia articles and the answer to the question is guaranteed to be a span in the context. Note here we use the SQuAD version 1.1 dataset, rather than SQuAD 2.0, since we do not focus on the unanswerable question detection problem.
- **MS MARCO** [22] is a large scale real-world RC dataset where the questions are collected from anonymous user logs from the Bing search engine. MS MARCO is not a typical RC dataset, since each question is paired with ten candidate passages which need an additional ranking step. In most existing works[19, 43], MS MARCO has been formed as an extractive neural RC dataset where the ground truth answer is typically defined as the span that has the max overlap with the human-written answer.
- **CoQA** [29] is also a variant dataset for the RC study. Different from the SQuAD dataset, CoQA contains sequentially dependent question-answer pairs in each context passage. Moreover, it requires the annotators to highlight the evidence in the context passage and provide a natural language answer to the question.

The detailed statistics of these datasets are shown in Table 1. We can see that these datasets differ with each in the length of context passsage, ranging from 56 terms to 270 terms. All these datasets contain more than 90,000 instances which can well support the development of neural RC models.

4.1.2 Evaluation Metrics. For evaluation, we take F_1 score and Exact Match (EM) as the metrics. F_1 score measures the average term-level overlap between the predicted answer and the ground truth answer. Specifically, given a predicted answer $\hat{A} = \{c_{\hat{s}}, \ldots, c_{\hat{e}}\}$ and its ground truth answer $A^* = \{c_{s^*}, \ldots, c_{e^*}\}$, F_1 is defined as follows

$$Precision = \frac{|\hat{A} \cap A^*|}{|\hat{A}|}, \tag{6}$$

$$Recall = \frac{|\hat{A} \cap A^*|}{|A^*|}, \tag{7}$$

$$F_1 = \frac{2 * Precision * Recall}{Precision + Recall}. \tag{8}$$

EM measures the percentage of the exact extraction of the ground truth answer,

$$\text{EM} = \mathbb{I}(A^* = \hat{A}).$$

Besides, for the MS MARCO dataset, since the answers are usually long, and the term order matters. Thus, we follow the previous work [19, 43] to take the ROUGE-L [18] as the evaluation metric:

$$R_{lcs} = \frac{LCS(A^*, \hat{A})}{|A^*|}, \tag{9}$$

$$P_{lcs} = \frac{LCS(A^*, \hat{A})}{|\hat{A}|}, \tag{10}$$

$$ROUGE - L = \frac{(1 + \beta^2) R_{lcs} P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}}, \tag{11}$$

where LCS is the longest common sequence algorithm which measures the overlaps between two sequences.

4.1.3 Baselines. We compare our LD-MLE with existing learning criteria for the RC task, including the maximum likelihood method, and two existing RL-based methods.

- **MLE** denotes the maximum likelihood estimation objective widely used in most neural RC models.
- **SCST** is a RL-based objective which is proposed in DCN+ [42] for RC. It optimizes the expected reward (i.e., the expected F_1 score) by sampling from model output distribution. To overcome the unstable nature of the RL methods, it needs to be combined with the MLE objective while in learning.
- **DCRL** is another RL-based objective for RC proposed in R.M-Reader [11]. It tackles the convergence suppression problem in SCST through dynamic-critical reinforcement learning. Note that DCRL also needs to combine the RL-based objective with MLE to keep learning stable.

4.1.4 RC models. We test the above training objectives based on three representative neural RC models. We will introduce these models in detail under the four layers framework, as is described in section 2.1.

- **BiDAF** [30] introduces bi-direction attention to improve question-context interaction. Concretely, character-level embedding and pre-trained word-level embeddings are concatenated to represent the word. These word embedding are passed to the encoding layer, which is implemented as an LSTM layer. In the interaction layer, the bi-direction attention mechanism is used to interact the context with the question in bi-direction. BiDAF is a representative RC model aiming to improve the interaction layer.
- **SAN** [20] proposes a stochastic answer network for iterative reasoning. SAN uses a similar layer structure with the BiDAF except for the answer layer. In the answer layer, an iterative answer module conducts multiple reasoning and produces a series of probability distributions of the start and end positions. These distributions are averaged to form the final distribution. While in training, they also applied a dropout layer on the outputs. SAN is a representative RC model aiming to improve the answer layer.
- **BERT** [4] is a multi-layer Transformer encoder based on the original implementation described in Vaswani et al. [36] and pre-trains parameter based on the masked language object and next sentence prediction task. Individually, on RC task, it packs the question and the context as a whole sequence, and encode the sequence using BERT encoder with the pre-trained parameters. Then a classification layer is applied on the encoded representations to get the probability distributions of the start and end positions. BERT is a representative work involving transfer learning in the RC.

4.1.5 Implementation Details. We use the official implementation of BiDAF [1] , SAN [2] , and BERT [3] from their original authors, and we apply different learning objectives on them. We train these models

[1] https://github.com/allenai/allennlp
[2] https://github.com/kevinduh/san_mrc
[3] https://github.com/google-research/bert

Table 2: Overall results on the SQuAD 1.1 dataset. The improvements of all the methods over MLE are shown in the brackets. $*$ and $**$ indicates statistically significance with p-value < 0.05 and p-value<0.01, respectively.

Metrics	Methods	BiDAF	SAN	BERT
F_1	MLE	79.6	84.1	88.1
	SCST	79.7*(0.13%)	84.5*(0.48%)	88.6*(0.57%)
	DCRL	79.7*(0.13%)	83.8(-0.36%)	88.6*(0.57%)
	LD-MLE	**80.2******(+0.75%)**	**84.6******(+0.60%)**	**89.0******(+1.02%)**
EM	MLE	70.2	76.2	80.8
	SCST	70.3*(0.14%)	76.8**(0.79%)	81.2*(0.5%)
	DCRL	70.3*(0.14%)	76.0*(-0.26%)	81.4*(0.74%)
	LD-MLE	**71.0******(+1.14%)**	**77.3******(+1.44%)**	**82.0******(+1.49%)**

Table 3: Results of BERT on MS MARCO and CoQA datasets.

Methods	MS MARCO	CoQA	
	Rouge-L	EM	F_1
MLE	47.3	69.5	78.8
SCST	47.8**(+1.10%)	70.0*(0.72%)	79.2*(+0.51%)
DCRL	47.9** (+1.20%)	70.1*(0.86%)	79.2*(+0.51%)
LD-MLE	**48.2******(+1.90%)**	**70.6******(+1.58%)**	**79.7***$*$**(+1.14%)**

Table 4: Results of MLE, ULS-MLE, Gaussian-MLE and LD-MLE methods on SQuAD dataset based on BERT.

	EM	△EM	F_1	△F_1
MLE	80.8	-	88.1	-
ULS-MLE	81.0	+0.25%	88.5	+0.45%
Gaussian-MLE	81.2	+0.5%	88.4	+0.34%
LD-MLE	**82.0**	**+1.45%**	**89.0**	**+1.02%**

on the training set and report the result on the dev set as in most previous works [20, 30]. Note that we don't report the result of test sets of these datasets, as we aim to verify the effectiveness of the learning objective which can be applied to almost all existing RC model. The test sets of these datasets are not available and we do not submit the system to the leaderboard system. We use Adam [14] optimizer with the learning rate set as 5e-5. Other hyper-parameters keep the same as the original implementation. We re-implement SCST and DCRL based on the description in DCN+ [42] and R.M-Reader [11]. All the code will be released soon after the anonymous review.

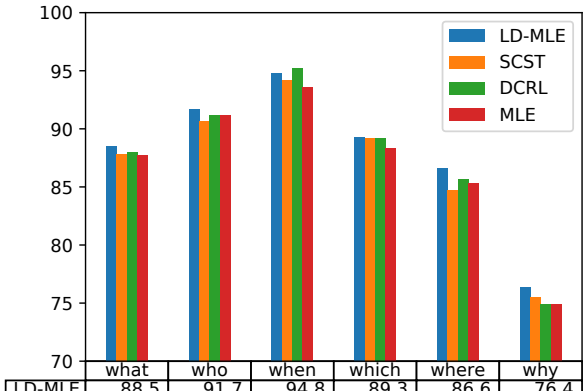
4.2 Main Results

4.2.1 Comparisons on the Basic RC Task. The main results on the SQuAD dataset are summarized in Table 2. As we can see: 1) The BERT model performs the best compared with the other models in terms of both F_1 and EM metrics. This demonstrates the power of the deep contextual representation learning in RC task. 2) The RL-based criteria (i.e., SCST and DCRL) can outperform the basic MLE criterion on some RC models. However, we also observe that the performance of the RL-based criteria drops below the MLE method sometimes. For example, the performance of SAN with the DCRL criterion is slightly lower than that with the MLE criterion. This may be due to the fact that RL-based methods rely on sampling from the non-stationary distribution which is often unstable in learning [23]. 3) The LD-MLE method can achieve consistently better results on all the four RC models as compared with all the baseline criteria. All the improvements of LD-MLE criterion against the MLE criterion are statistically significant (p-value < 0.01). For example, BERT achieves 1.49% and 1.02% performance improvement in terms of EM and F_1 respectively when the learning objective changes from MLE to LD-MLE. 4) Although our LD-MLE objective is derived based on the F_1 metric, it is interesting to see that our method can also improve the EM metric with a substantial margin. For example, the improvement of SAN and BERT with LD-MLE against MLE is about 1.44% and 1.49% in terms of EM, respectively. This demonstrates that LD-MLE could improve the generalization ability of the RC model. All these results demonstrate the effectiveness of our LD-MLE criterion.

4.2.2 Comparisons on the Variant RC Tasks. To investigate how the LD-MLE method performs on different types of RC tasks, we conduct experiments on two variants of RC tasks, namely multi-passage RC task (i.e., MS MARCO dataset) and conversational RC task (i.e., CoQA dataset). Here we only take BERT as the RC model since BERT is the state-of-the-art model on these datasets. The results are shown in Table 3. We can see RL-based methods have very close results on the two variant tasks, which are slightly better than the MLE criterion. Moreover, the LD-MLE method can achieve the best performance among all the criteria, and can significantly outperform (i.e., p-value < 0.01) the MLE criteria in terms of all the evaluation metrics over the two variant RC tasks. The results indicate that although the LD-MLE method is proposed for the basic RC task, it can also help improve more complicated RC tasks by taking into account the structure of the output space. It is worth to note that the MS MARCO dataset and CoQA dataset contain more noises compared with SQuAD dataset, which demonstrate that our LD-MLE is robust as it improves the performances consistently over MLE on all three datasets.

4.3 Detailed Analysis

Question: How about applying other types of label distribution?



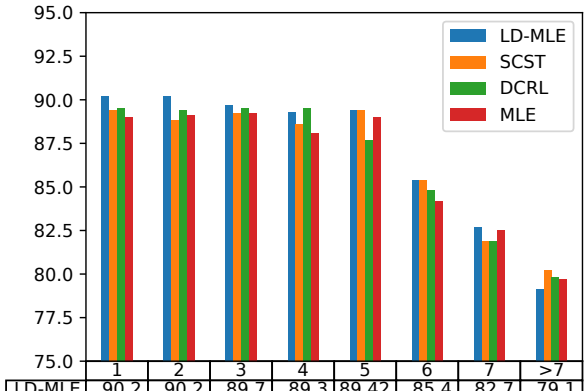
	what	who	when	which	where	why
LD-MLE	88.5	91.7	94.8	89.3	86.6	76.4
SCST	87.8	90.7	94.2	89.2	84.7	75.5
DCRL	88.0	91.2	95.2	89.2	85.7	74.9
MLE	87.7	91.2	93.6	88.3	85.3	74.9

(a) Performance breakdown by question type with BERT.

Question Type	what	who	when	which	where	why
Percentage	57.46	11.75	6.78	6.09	4.36	1.42

(b) Statistics of different question types.

Figure 3: Performance comparison over different question types on SQuAD dataset.



	1	2	3	4	5	6	7	>7
LD-MLE	90.2	90.2	89.7	89.3	89.42	85.4	82.7	79.1
SCST	89.4	88.8	89.2	88.6	89.4	85.4	81.9	80.2
DCRL	89.5	89.4	89.5	89.5	87.7	84.8	81.9	79.8
MLE	89.0	89.1	89.2	88.1	89.0	84.2	82.5	79.7

(a) Performance breakdown by answer length with BERT.

Answer length	1	2	3	4	5	6	7	>7
Percentage	32.4	27.2	16.0	8.0	4.7	2.9	2.0	6.7

(b) Statistics of different answer lengths.

Figure 4: Performance comparison over different answer length on SQuAD dataset.

In the LD-MLE criterion, the label distribution is defined based on the evaluation metric (i.e., F_1 metric). As discussed in previous section, our method is related to the label smoothing technique. Here, we try to investigate different heuristic distributions rather than the evaluation metric in the LD-MLE. We consider two alternative distributions over the labels as the target distribution. Specifically, the first one is to directly take part of the probability from the ground-truth label, and uniformly assign it to each term in the span. We name this method as Uniform Label Smoothing-based MLE (ULS-MLE for short). The second one utilizes the Gaussian distribution to smooth the probability of ground truth, where the mean is the ground truth label and the variance is a hyper-parameter to be tuned. We name this method as Gaussian-MLE. The results are shown in Table 4. From the results, we can see that the ULS-MLE and Gaussian-MLE can slightly improve the performance against MLE criterion. This is due to the fact that smoothing target distribution could increase the model's generalization ability [34]. Moreover, the LD-MLE, which incorporates the evaluation metric in the learning objective, obtains the best performance against all the baseline methods. The results indicate that it is more effective to leverage evaluation-dependent task reward to define the target distribution.

Question: How does the model perform under different question types?

To examine how the LD-MLE criterion performs in terms of different types of questions, we divide the development set of SQuAD by question type based on their respective WH-words, such as "what" and "when". The F_1 scores and statistics of question types are shown in Figure 3. From the results, we can see that RL-based methods can achieve good performance on a few question types. For example, the SCST can outperform MLE on question type "when", "which", and "why". However, there are some types of question that RL-based methods are less effective than the MLE method, e.g., the question type "where" and "who". For our LD-MLE criterion, we can observe that it can consistently outperform MLE on all the types of questions. This result demonstrates that LD-MLE is more stable compared with the RL-based methods. From the statistics of frequency, "what"-type question and "who"-type question are about 57% and 11% among the total. LD-MLE outperforms RL-based method on these major type questions, which shows the advantage of the LD-MLE criterion in capturing the major patterns in the data.

Question: How does the model perform under different lengths of answers?

Here, we aim to understand the impact of the answer length on different training criteria. We conduct experiments on SQuAD dataset with all baseline methods. Here we only show the results of F_1 metric as the EM metric will obtain similar conclusions. The statistics of the answer length are shown in Figure 4 (b). We can see that most of the answers are very short, e.g., there are about 75.6% of the answers with less than three terms. The $F1$ results are shown in the Figure 4 (a). We can see that in general the RL-based criteria can achieve better performance than the MLE criteria on questions with different lengths of answers, but may fail at some long answers, e.g., the answer length at 5 or 7. Moreover, the LD-MLE criterion obtains larger improvements than RL-based methods against MLE on questions with shorter answer spans. When the answer length becomes longer, the improvements reduce accordingly. For answers with length more than 7 terms, the LD-MLE criterion may also

becomes slightly worse than the MLE criterion. A possible reason is that the LD-MLE criterion will use much smoother distribution when the answer span becomes longer. In this way, it may suffer from the loss of discrimination between the ground truth answer span and other structure proximal spans. Therefore, further considering the answer length information in the label distribution may help improve the performance of LD-MLE, and we leave it as our future work.

5 RELATED WORK

In this section, we first review previous works on improving the model structures of the RC task. We then describe existing criteria for learning RC models and discuss their connections and differences with LD-MLE.

5.1 Reading Comprehension Model

Generally, the RC models consist of four conceptual layers, namely embedding layer, encoding layer, interaction layer, and answer layer. Most of existing works devoted their effort on improving the interaction layer or the answer layer to achieve better performance. Thus, we will describe these works in these two directions.

For improving *the interaction layer*, most works focus on the attention mechanism to better model the question and passage interaction. For example, Wang and Jiang [38] proposed conditional attention mechanisms in the Match-LSTM encoder. BiDAF [30] applied bidirectional attention flow to attend to the question and document simultaneously. RNet [40] further introduced the self-attention mechanism on the context passage and enhanced the LSTM with gating mechanism to amplify useful information. FusionNet [12] and SLQA [39] incorporated hierarchical attention mechanism to interact multi-level representations of the question and context passage. They also applied a complex fusion function to combine the attended vectors and encoded vectors.

For improving *the answer layer*, the works aim to improve the answer prediction step based on the passage representation. SAN [20] is a typical work which proposed to iteratively predict multiple distributions and drop some of them to get the final result. ReasoNet [32] made use of multi-turn reasoning module to iteratively find the answer from the context passage with reinforcement learning. In this way, the ReasoNet can dynamically determine whether to continue the reasoning or to terminate reading. These works usually make multiple predictions to correct the error in single prediction.

There are also another line of works which employ transfer learning techniques to improve the RC models, such as CoVE [21], ELMo [25], GPT [26], and BERT [4]. These models pre-trained part of the parameters on other tasks with large corpus, e.g., various language modeling tasks or the machine translation task. They have achieved a great success on many natural language processing tasks recently [37, 41]. Especially, ELMo pretrained the bi-direction RNN language model and is used as auxiliary contextual embeddings in existing models [12, 20]. The BERT model, which pretrained the parameters on a novel mask language model, has achieved state-of-the-art performance on the RC task [28].

In this paper, we select three representative RC models which have focused on improving different part of the RC model, i.e., BiDAF mainly improved the interaction layer, SAN improved the answer layer, and BERT incorporated the transfer learning techniques. We apply our LD-MLE objective and other baseline objectives over these models to validate the effectiveness of the proposed criterion.

5.2 Learning Criteria for RC Task

Most existing RC models relied on the MLE criterion for the optimization, such as Fusionnet [12] ,DrQA [1] and documentqa [2]. They trained the model output to maximize the ground-truth position while minimizing all alternative outputs, even if they are similar to the ground-truth answer.

One way to improve the MLE is to incorporate the evaluation metric into the training objective. This line of works includes minimal risk training [31], maximum expected reward [9, 17, 24], and expected loss optimization [45]. These methods have been explored in the sequence to sequence model and certain computer vision task. There are also a few works attempt to address these problem in the RC task [11, 42]. For example, DCN+ model [42] proposed a mixed objective that combines cross-entropy loss and self-critical policy learning derived from word overlap to improve MLE. R.M-reader [11] introduced dynamic-critical reinforcement learning to further address the convergence suppression problem occurred in DCN+ model. However, these RL-based methods suffer from training inefficiency as they relied on sampling from the model output. They have to both append the original MLE loss to keep the training process stable. In this work, we improves MLE in a simple and stable way by taking an auxiliary label distribution in the MLE framework.

6 CONCLUSION

In this paper, we proposed a new learning criterion LD-MLE for the RC task by taking into account the structure of the output space. This learning criterion can be applied over almost all the existing RC models to improve their optimization process. Our experimental results demonstrated the effectiveness of our LD-MLE method over the traditional MLE and RL-based methods. We encourage future neural RC models to use this criterion to replace the simple MLE for better model learning. In future work, we will try to investigate other label distributions on the learning performance.

7 ACKNOWLEDGMENTS

This work was supported by Beijing Academy of Artificial Intelligence (BAAI), and funded by the National Natural Science Foundation of China (NSFC) under Grants No. 61425016, 61722211, 61773362, 61872338, and 61902381, the Youth Innovation Promotion Association CAS under Grants No. 20144310, and 2016102, the National Key RD Program of China under Grants No. 2016QY02D0405, and the Foundation and Frontier Research Key Program of Chongqing Science and Technology Commission (No. cstc2017jcyjBX0059).

REFERENCES

[1] Danqi Chen. 2018. *Neural Reading Comprehension and Beyond.* Ph.D. Dissertation. Stanford University.
[2] Christopher Clark and Matt Gardner. 2017. Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723* (2017).

[3] Zihang Dai, Qizhe Xie, and Eduard Hovy. 2018. From Credit Assignment to Entropy Regularization: Two New Algorithms for Neural Sequence Prediction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1672–1682.

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

[5] Maha Elbayad, Laurent Besacier, and Jakob Verbeek. 2018. Token-level and sequence-level loss smoothing for RNN language models. *arXiv preprint arXiv:1805.05062* (2018).

[6] Bin-Bin Gao, Chao Xing, Chen-Wei Xie, Jianxin Wu, and Xin Geng. 2017. Deep label distribution learning with label ambiguity. *IEEE Transactions on Image Processing* 26, 6 (2017), 2825–2838.

[7] Xin Geng. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering* 28, 7 (2016), 1734–1748.

[8] Xin Geng, Chao Yin, and Zhi-Hua Zhou. 2013. Facial age estimation by learning from label distributions. *IEEE transactions on pattern analysis and machine intelligence* 35, 10 (2013), 2401–2412.

[9] Xiaodong He and Li Deng. 2012. Maximum expected bleu training of phrase and lexicon translation models. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 292–301.

[10] Karl Moritz Hermann, 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*. 1693–1701.

[11] Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. 2017. Reinforced mnemonic reader for machine reading comprehension. *arXiv preprint arXiv:1705.02798* (2017).

[12] Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2017. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. *arXiv preprint arXiv:1711.07341* (2017).

[13] Mandar Joshi, Eunsol Choi, Daniel 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)*. 1601–1611.

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] Vijay R Konda and John N Tsitsiklis. 2000. Actor-critic algorithms. In *Advances in neural information processing systems*. 1008–1014.

[16] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 785–794.

[17] Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016. Deep Reinforcement Learning for Dialogue Generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 1192–1202.

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

[19] Jiahua Liu, Wan Wei, Maosong Sun, Hao Chen, Yantao Du, and Dekang Lin. 2018. A Multi-answer Multi-task Framework for Real-world Machine Reading Comprehension. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2109–2118.

[20] Xiaodong Liu, 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)*. 1694–1704.

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

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

[23] Mohammad Norouzi, Samy Bengio, Navdeep Jaitly, Mike Schuster, Yonghui Wu, Dale Schuurmans, et al. 2016. Reward augmented maximum likelihood for neural structured prediction. In *Advances In Neural Information Processing Systems*. 1723–1731.

[24] Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, 160–167.

[25] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 2227–2237. https://doi.org/10.18653/v1/N18-1202

[26] Alec Radford, 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/research-covers/languageunsupervised/language understanding paper. pdf* (2018).

[27] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You DonâĂŹt Know: Unanswerable Questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 784–789.

[28] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2383–2392.

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

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

[31] Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum Risk Training for Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 1683–1692.

[32] Yelong Shen, 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*. ACM, 1047–1055.

[33] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*. 1057–1063.

[34] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.

[35] Adam Trischler, 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*. 191–200.

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

[37] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461* (2018).

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

[39] Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-Granularity Hierarchical Attention Fusion Networks for Reading Comprehension and Question Answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1705–1714.

[40] Wenhui Wang, 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)*. 189–198.

[41] Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 1112–1122.

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

[43] Ming Yan, Jiangnan Xia, Chen Wu, Bin Bi, Zhongzhou Zhao, Ji Zhang, Luo Si, Rui Wang, Wei Wang, and Haiqing Chen. 2018. A Deep Cascade Model for Multi-Document Reading Comprehension. *arXiv preprint arXiv:1811.11374* (2018).

[44] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600* (2018).

[45] Alan Yuille and Xuming He. 2012. Probabilistic models of vision and max-margin methods. *Frontiers of Electrical and Electronic Engineering* 7, 1 (2012), 94–106.