# 基于多任务学习的生成式阅读理解

**钱锦**[1]，**黄荣涛**[1]，**邹博伟**[1,2]*，**洪宇**[1]
苏州大学计算机科学与技术学院，苏州215000[1]
新加坡资讯通信研究院，新加坡138632[2]
{jaytsien,rthuang.suda}@gmail.com, zou_bowei@i2r.a-star.edu.sg,
tianxianer@gmail.com

## 摘要

生成式阅读理解是机器阅读理解领域一项新颖且极具挑战性的研究。与主流的抽取式阅读理解相比，生成式阅读理解模型不再局限于从段落中抽取答案，而是能结合问题和段落生成自然和完整的表述作为答案。然而，现有的生成式阅读理解模型缺乏对答案在段落中的边界信息以及对问题类型信息的理解。为解决上述问题，本文提出一种基于多任务学习的生成式阅读理解模型。该模型在训练阶段将答案生成任务作为主任务，答案抽取和问题分类任务作为辅助任务进行多任务学习，同时学习和优化模型编码层参数；在测试阶段加载模型编码层进行解码生成答案。实验结果表明，答案抽取模型和问题分类模型能够有效提升生成式阅读理解模型的性能。

**关键词：** 多任务学习 ；生成式阅读理解

# Generative Reading Comprehension via Multi-task Learning

**Jin Qian**[1],**Rongtao Huang**[1],**Bowei Zou**[1,2]*,**Yu Hong**[1]
School of Computer Science and Technology, Soochow University, Suzhou, 215000[1]
Institute for Infocomm Research, Singapore, 138632[2]
{jaytsien,rthuang.suda}@gmail.com, zou_bowei@i2r.a-star.edu.sg,
tianxianer@gmail.com

## Abstract

Generative reading comprehension is a novel and challenging research in the field of machine reading comprehension. Compared with the mainstream extractive reading comprehension, generative reading comprehension model is no longer limited to extract answers from paragraphs, but can combine questions and paragraphs to generate natural and complete statements as answers. However, the existing generative reading comprehension model lacks the understanding of the boundary information of answers in paragraphs and the question type information. To solve such issues, this paper proposes a generative reading comprehension model based on multi-task learning. In the training phase, the model takes the answer generation task as the main task, and the answer extraction and question classification tasks as auxiliary tasks for multi-task learning. The model simultaneously learns and optimizes the parameters of the model encoding layer. Then it loads the encoding layer in the test phase to decode and generate the answers. The experimental results show that the answer extraction model and the question classification model can effectively improve the performance of the generative reading comprehension model.

**Keywords:** Multi-task Learning , Generative Reading Comprehension

* 通讯作者

作为模型输入。每个词项表示为词向量$\mathbf{WE}(w_i)$、段向量$\mathbf{SE}(w_i)$和位置向量$\mathbf{PE}(w_i)$的和，维度均为$d_w$，其中词向量用于表示不同词项，段向量用于区分词来自源序列还是目标序列，位置向量用于表示词在输入序列中的绝对位置。词向量$\mathbf{X}_i$表示为：

$$\mathbf{X}_i = \mathbf{WE}(w_i) + \mathbf{SE}(w_i) + \mathbf{PE}(w_i)$$

其中$w_i$为第$i$个位置的词项。

本文将词向量集合表示为$\{\mathbf{X}_i\}_{i=1}^{|x|}$，则输入序列表示为$\mathbf{H}^0 = [\mathbf{X}_1, ..., \mathbf{X}_{|x|}]$，其中$|x|$为输入序列的长度。UniLMv2的编码层使用12层堆叠的Transformer网络，每经过一层Transformer网络都能得到不同抽象层次的上下文表示：

$$\mathbf{H}^l = [h_1^l, ..., h_{|x|}^l] = Transformer_l(\mathbf{H}^{l-1}), l \in [1, 12] \tag{2}$$

其中$l$为第$l$层Transformer网络，$h_i^l$为第$i$个词项的$l$层隐层表示。

Tranformer网络由多头自注意力机制和前向神经网络两个子层组成，每个子层均使用残差连接和层正则化，因此每个子层的输出可表示为：

$$LayerNorm(x + Sublayer(x))$$

第$l$层Transformer网络的自注意力头$\mathbf{A}_l$计算如下：

$$\mathbf{A}_l = softmax(\frac{\mathbf{Q}_l\mathbf{K}_l^\mathsf{T}}{\sqrt{d_k}} + \mathbf{M})\mathbf{V}_l \tag{3}$$

$$\mathbf{Q}_l = \mathbf{H}^{l-1}\mathbf{W}_l^Q, \mathbf{K}_l = \mathbf{H}^{l-1}\mathbf{W}_l^K, \mathbf{V}_l = \mathbf{H}^{l-1}\mathbf{W}_l^V \tag{4}$$

$$\mathbf{M}_{i,j} = \begin{cases} 0, & \text{允许被注意} \\ -\infty, & \text{不允许被注意} \end{cases} \tag{5}$$

其中$\mathbf{Q}_l$，$\mathbf{K}_l$和$\mathbf{V}_l$分别代表第$l$层注意力机制中的查询（query）向量、键（key）向量和值（value）向量，$d_k$为向量$\mathbf{K}_l$的维度，$\mathbf{W}_l^Q, \mathbf{W}_l^K, \mathbf{W}_l^V \in \mathbb{R}^{d_h \times d_k}$为可学习参数矩阵，$\mathbf{H}^{l-1} \in \mathbb{R}^{|x| \times d_h}$为$l-1$层的隐层表示，$\mathbf{M}$为生成式阅读理解模型注意力遮蔽矩阵，如图2所示。

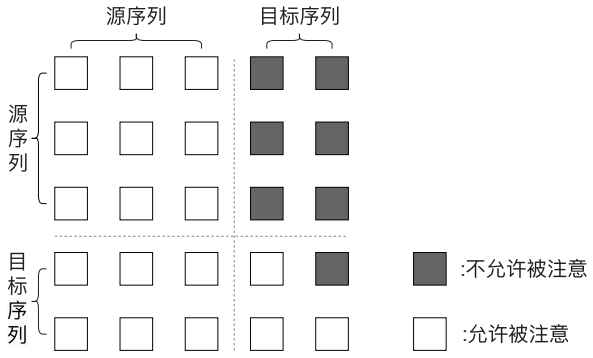


图 2: 注意力遮蔽矩阵

通过上述词嵌入层和Tranformer网络，得到输入序列的上下文表示$\mathbf{H}^1, \mathbf{H}^2, ..., \mathbf{H}^{12}$。本文使用最后一层输出$\mathbf{H}^{12}$作为整个序列的表示。$\mathbf{H}^{12}$中包含问题、段落和答案表示，其中，段落表示部分记作$\mathbf{H}^p$，答案表示部分记作$\mathbf{H}^a$，问题类别表示记作$\mathbf{H}^{cls}$。根据图 2所示的注意力遮蔽矩阵可知，问题和段落不会和答案进行交互，保证了训练和测试阶段$\mathbf{H}^p$和$\mathbf{H}^{cls}$所含信息的一致性。

### 3.3 任务层

作为基于多任务学习框架的核心部分，任务层由答案生成模型、答案抽取模型和问题分类模型三部分构成。

**答案生成模型** 训练阶段，真实答案会以一定概率被随机遮蔽，并且同时保留其原始位置信息来实现部分自回归（随机预测答案被遮蔽的片段），答案中被遮蔽的词项在经过编码后得到答案表示$\mathbf{H}^a$。答案生成模块通过解码层对原始答案中被遮蔽的词项进行预测来生成答案。具体来说，$\mathbf{H}^a$首先经过线性层并用Gelu函数激活后进行层正则化：

$$\mathbf{H}^a = LayerNorm(Gelu(Linear(\mathbf{H}^a))) \tag{6}$$

然后通过线性层将每个被遮蔽的词项映射到模型词表空间，获得预测分数。最后，使用SoftMax函数计算词的概率向量$\mathbf{a}$：

$$\mathbf{a} = SoftMax(Linear(\mathbf{H}^a)) \tag{7}$$

本文采用有标签平滑优化的交叉熵损失函数计算答案生成模型的目标函数：

$$L_{generate} = \sum_{t=1}^{T}\sum_{k=1}^{K} y_{\mathbf{a}_t^k} \cdot \log \mathbf{a}_t^k \tag{8}$$

其中$T$表示答案的长度，$K$表示词表的大小，$y_{\mathbf{a}_t^k}$表示答案中第t个位置经过标签优化的真实标签，$\mathbf{a}_t^k$表示答案中第t个位置的预测标签。注意，本文只对答案中被遮蔽的词项计算损失。

测试阶段，模型对于输入的问题和段落，每个时间步经解码层预测当前词的生成概率，同时使用束搜索每次保留生成概率最大的前k个序列，直至模型预测出[EOS]终止符结束解码。最后，模型将束搜索结果中生成概率最大的序列解码输出，其概率计算为：

$$P(A|Q,P) = P(a_1|Q,P)P(a_2|Q,P,a_1)...P([EOS]|Q,P,a_1,a_2,...) \tag{9}$$

**答案抽取模型** 经过编码层后，段落被表示为矩阵$\mathbf{H}^p$，答案抽取模型通过指针网络对答案的起始和终止位置进行识别。具体地，$\mathbf{H}^p$分别经过线性层得到对应起始位置分数和终止位置的分数，并通过SoftMax函数对分数进行归一化，得到相应的概率向量：

$$\mathbf{s}, \mathbf{e} = SoftMax(Linear(\mathbf{H}^p)) \tag{10}$$

其中$\mathbf{s}$为预测答案的起始位置概率向量，$\mathbf{e}$为答案终止位置概率向量，$\mathbf{s}$和$\mathbf{e}$由不同参数的线性层计算得到。

本文采用交叉熵损失函数计算答案抽取模型的目标函数：

$$L_{extract} = y_{\mathbf{s}} \cdot \log \mathbf{s} + y_{\mathbf{e}} \cdot \log \mathbf{e} \tag{11}$$

其中$y_{\mathbf{s}}$表示真实答案的起始位置概率向量，$y_{\mathbf{e}}$表示真实答案的终止位置概率向量。

**问题分类模型** 由于CoQA数据集中存在多种问题类型，包括事实型问题（Factoid question）、是非类问题（True/False question）和不可回答问题（Unanswerable question）。针对不同类型的问题，答案的模式通常差别较大，例如针对是非类问题，答案通常以“Yes/No”开头。本文采用4种问题类型标签{0: yes; 1: no; 2: unanswerable; 3: factoid}，以上四种问题类型（其中是非类问题被分为两种不同类型）。如图 1所示，输入经过编码后，取出[CLS]表示用于获得问题类型表示即$\mathbf{H}^{cls}$，并经过线性层为问题类型打分，最后将分数进行归一化后形成分类概率：

$$\mathbf{c} = SoftMax(Linear(\mathbf{H}^{cls})) \tag{12}$$

其中，$\mathbf{c}$代表问题类型的分数向量。

本文采用交叉熵损失函数计算问题分类模型的目标函数：：

$$L_{cls} = \sum_{k=1}^{K} y_{\mathbf{c}^k} \cdot \log \mathbf{c}^k \tag{13}$$

其中$K = 4$表示问题类别数，$y_{\mathbf{c}^k}$表示真实类别标签，$\mathbf{c}^k$表示预测类别标签。

**多任务学习** 本文采用多任务学习的方法，在训练阶段同时学习和更新答案生成、答案抽取和问题分类模块共享的编码层参数，让答案抽取和问题分类任务辅助答案生成任务提升阅读理解模型的性能。模型的损失由生成损失、抽取损失和分类损失三部分共同组成，整个模型的目标函数为：

$$LOSS = L_{generate} + \lambda_1 L_{extract} + \lambda_2 L_{cls} \tag{14}$$

其中$\lambda_1$和$\lambda_2$为调和系数，用于调节辅助任务权重。

## 4 实验

本章首先介绍生成式阅读理解任务数据集和实验设置，然后报告本文提出的基于多任务的生成式阅读理解模型性能，并针对实验结果进行分析。

### 4.1 生成式阅读理解任务数据集

现有阅读理解数据集大多针对抽取式模型，即答案为篇章中的一个片段，如SQuAD (Rajpurkar et al., 2016)、HotpotQA (Yang et al., 2018)等。采用这些数据集无法全面评价生成式阅读理解模型，与抽取式模型相比，其在答案的可读性、表述的完整性、以及应对多段答案的问题上，均有较大区别(请见本文第一章)。基于上述原因，本文实验中采用以下三个数据集。

- CoQA(Conversational Question Answering)[2]

  CoQA基于多个领域的多轮对话进行构建，并保持了人类对话简短的特征，存在大量指代和省略现象，问题和答案普遍偏短。值得注意的是，为了保证该数据集尽可能贴近自然对话，其中78%的答案经过人工编辑；此外，该数据集中存在较多的是非类问题(19.8%)和不可回答问题(1.3%)，部分问题无法仅采用抽取式阅读理解模型回答 (Reddy et al., 2018)。尽管如此，目前在CoQA评测榜单上排名较高的均为抽取式模型，而生成式模型，如UniLM和ERNIE-GEN，仅报告了在验证集上的性能，因此，本文将CoQA的验证集作为测试集评价系统性能，调参使用的验证集从CoQA训练集中划分。

- MS MARCO(Microsoft Machine Reading Comprehension)[3]

  MS MARCO是一个多文档问答数据集 (Bajaj et al., 2018)，其中特别提供了一个自然语言生成(NLG)子数据集，该数据集由人工编辑答案，其答案并非严格匹配文档中的片段，因此，本文采用MS MARCO(NLG)作为评价生成式阅读理解模型的数据集。注意，由于该数据集还包含了文档检索任务，而本文研究重点仅在于机器阅读理解，因此，仅采用人工编辑答案时依据的文档，即最佳文档(golden passage)；此外，由于MS MARCO评测榜单上，NLG数据集同样包含了文档检索任务，因此本文仅报告模型在MS MARCO(NLG)验证集上的结果。

- NarrativeQA[4]

  NarrativeQA是一个生成式阅读理解数据集，该数据集基于书本故事和电影脚本构建，答案由人工编辑 (Kočiský et al., 2018)。本文基于数据集的摘要子集进行阅读理解，并在其测试集上进行测试。

表 2列出了本文所采用三个数据集的统计数据。CoQA中存在28.7%的命名实体类问题、19.6%的名词短语类问题和9.8%的数字类问题；NarrativeQA中存在30.54%的人名类问题、9.73%的地点类问题和约10%左右的事件、实体、数字类问题，且CoQA和NarrativeQA明确允许简短、自然的答案，因此CoQA和NarrativeQA的答案普遍较短。MS MARCO(NLG)中存在53.12%的描述型问题，且答案会融入问题信息形成完整的表述，答案普遍较长。

### 4.2 实验设置

本文使用的模型为微软开源的unilm1.2-base-uncased[5]，该模型在大多数自然语言生成任务上取得了最佳性能。针对不同数据集，表 3列出了模型使用的超参数设置。

[2]https://stanfordnlp.github.io/coqa/
[3]https://microsoft.github.io/msmarco/
[4]https://github.com/deepmind/narrativeqa
[5]https://unilm.blob.core.windows.net/ckpt/unilm1.2-base-uncased.bin

表 2: CoQA、MS MARCO和NarrativeQA数据集(#问题数)

| 数据集 | CoQA | MS MARCO(NLG) | NarrativeQA(Summary) |
|---|---|---|---|
| 训练集# | 108,647 | 153,725 | 32,747 |
| 验证集# | 7,983 | 12,467 | 3,461 |
| 测试集# | - | - | 10,557 |
| 段落平均长度 | 271 | 53 | 659 |
| 问题平均长度 | 5.5 | 6 | 9.83 |
| 答案平均长度 | 2.7 | 14 | 4.73 |

表 3: 参数设置

| 参数描述 | CoQA | MS MARCO | NarrativeQA |
|---|---|---|---|
| max_src_len | 470 | 176 | 470 |
| max_tgt_len | 42 | 40 | 42 |
| $\lambda_1$ | 0.245 | 0.1 | 0.1 |
| $\lambda_2$ | 1 | 0 | 0 |
| batch size | 48 | 48 | 48 |
| 学习率 | 2e-5 | 7e-5 | 2e-5 |
| Warmup率 | 0.1 | 0.02 | 0.1 |
| epoch | 10 | 2 | 10 |

在CoQA多轮对话数据集中，当前问题可能存在指代或省略现象，因此本文选取当前问题之前的至多两轮问答对作为对话历史，并与当前问题进行拼接当作完整的问题$Q$，同时使用上一轮答案和当前问题的词在段落中出现的频率选取文章中最佳的段落作为段落$P$。训练时，根据答案$A$计算出其在段落$P$中的起始位置和终止位置(答案不在段落中时，起始位置和终止位置均设为0)。实验中，问题最大长度设为60，问题和段落(源序列)的最大长度为470，答案(目标序列)的最大长度为42，该数据处理与 (Dong et al., 2019)论文里的方法保持一致。模型的优化器为AdamW。

在MS MARCO多文档阅读理解数据集中，每个问题$Q$会给定10个参考段落，本文直接选取最佳的段落进行拼接作为段落$P$。训练时，根据答案$A$计算出其在段落$P$中的起始位置和终止位置(答案不在段落中时，起始位置和终止位置均设为0)。实验中，问题和段落(源序列)的最大长度为176，答案(目标序列)的最大长度为40。模型的优化器为AdamW。

在NarrativeQA数据集中，本文使用问题$Q$的词在段落中出现的频率选取摘要中最佳的段落作为段落$P$。训练时，使用F1值选取段落$P$中与答案$A$最为接近的片段作为抽取答案，并根据抽取答案计算出答案$A$在段落$P$中的起始位置和终止位置。实验中，问题和段落(源序列)的最大长度为470，答案(目标序列)的最大长度为42。模型的优化器为AdamW。

本文在CoQA数据集上使用F1值 (Rajpurkar et al., 2016)来评价模型的性能，在MS MARCO和NarrativeQA数据集上使用BLEU (Papineni et al., 2002)和ROUGE-L (Lin, 2004)来评价模型的性能。

### 4.3 实验结果与分析

为了验证本文基于多任务的生成式阅读理解方法的有效性，本文与以下阅读理解模型进行了比较:

- **UniLM**：由Dong等 (2019)提出，是第一个在CoQA数据集上报告实验性能的预训练生成模型，本文在实验设置上和它保持一致。

- **ERNIE-GEN**：由Xiao等 (2020)提出的基于多流（multi-flow）机制生成完整语义片段的预训练生成模型，在CoQA生成式阅读理解中达到了目前最好的性能。

- **Masque**：由Nishida等 (2019)提出的多风格生成式阅读理解模型，在MS MARCO(NLG)和NarrativeQA数据集的相关指标上达到了目前的最好性能。
- **UniLMv2**：由Bao等 (2020)提出，采用伪遮蔽语言模型的预训练生成模型，是UniLM的改进版本。本文使用UniLMv2分别在三个数据集上进行实现作为我们的基线模型，并简单修复了wordpiece分词在解码时出现的分词错误。
- **MLT-Model**：本文提出的基于多任务学习的生成式阅读理解模型，由答案抽取和问题分类任务辅助生成式阅读理解模型。

表 4: 模型在CoQA验证集上的性能

| 模型 | F1 |
|---|---|
| UniLM | 82.5 |
| ENRIE-GEN | 84.5 |
| UniLMv2 | 86.1 |
| MLT-Model | **86.7** |

表 5: 模型在CoQA验证集的消融实验

| 模型 | F1 |
|---|---|
| MLT-Model | **86.7** |
| w/o cls | 86.0 |
| w/o extract | 86.2 |

表 4为本文提出的模型在CoQA验证集上的性能，我们的模型在F1指标上比当前性能最好的生成式模型ENRIE-GEN提升了2.2%，同时较基线模型UniLMv2提升了0.6%。本文针对预训练生成模型在答案解码时出现的子词结合不准确问题加以修复，实现的基线模型UniLMv2高于原始版本的性能，较ENRIE-GEN提升1.6%的F1值。表 5列出了本文模型在CoQA上的消融实验性能，在去除答案抽取任务和问题分类任务之后，性能较MTL-Model分别下降0.5%和0.7%的F1值。这是由于CoQA中存在20%左右的是非类问题和不可回答问题，这两类问题在训练阶段答案的起始和终止位置均设为0，因此仅用答案抽取任务辅助生成模型，会弱化模型对这两类问题的生成能力；而仅用问题分类任务来辅助生成模型，模型会缺少对答案在段落中边界信息的理解，所以只有将答案抽取和问题分类任务一起和答案生成任务进行多任务学习才能从整体上提升生成模型的性能。

表 6: 模型在MS MARCO(NLG)验证集(Golden Passage)上的性能

| 模型 | BLEU-1 | BLEU-4 | ROUGE-L |
|---|---|---|---|
| Masque | 78.14 | - | 78.80 |
| UniLMv2 | 79.76 | 68.87 | 80.09 |
| MLT-Model | **80.53** | **69.82** | **80.64** |

表 6为本文提出的模型在MS MARCO (NLG)验证集上选取最佳文档的性能表现。本文模型较基线模型UniLMv2在BLEU-1指标上提升0.77%，BLEU-4指标上提升0.95%，ROUGE-L指标上提升0.55%。这是由于MS MARCO(NLG)数据集中答案和选定段落中的部分片段相似度较高，答案抽取任务能够辅助模型关注答案在段落中的边界信息，并增强生成模型对问题和段落中答案片段之间关系的理解，最终提升生成模型的性能。我们在同样设置下和Masque模型进行了对比，本文所提模型在BLEU-1指标上提升了2.39%，ROUGE-L指标上提升了1.84%。这主要由于Masque模型仅使用静态的预训练词向量并基于Transformer网络进行答案生成，而本文模型基于网络更加复杂的预训练模型UniLMv2生成答案，因此在实验性能上取得较大提升。

表 7为本文模型在NarrativeQA(summary)测试集上的性能表现。本文模型较基线模型UniLMv2在BLEU-1指标上提升0.39%，BLEU-4指标上提升0.61%，ROUGE-L指标上提升0.1%。NarrativeQA数据集的答案长度普遍偏短，因此我们的模型并未在ROUGE-L指标上有明显提升，但是BLEU指标证明了答案抽取任务有助于生成模型生成更准确的答案。此外本文模型较目前性能最好的Masque模型在BLUE-1指标上提升了3.81%，BLEU-4指标上提升了1.24%，但在ROUGE-L指标上下降了0.53%。可能的原因是Masque模型基于整个摘要生成答案，而本文的模型是基于规则选取的滑窗作为段落来进行生成式阅读理解，在选取滑窗时丢失了部分性能；Masque模型在该数据集使用MS MARCO数据进行多风格学习，而本文模型并未

表 7: 模型在NarrativeQA(summary)测试集上的性能

| 模型 | BLEU-1 | BLEU-4 | ROUGE-L |
|---|---|---|---|
| Masque(NarrativeQA + MS MARCO) | 54.11 | 30.43 | **59.87** |
| Masque(NarrativeQA) | 48.70 | 20.98 | 54.74 |
| UniLMv2 | 57.53 | 31.06 | 59.24 |
| MLT-Model | **57.92** | **31.67** | 59.34 |

采用增加额外训练数据的方法训练模型。我们还比较了在相同训练数据的情况下，本文模型较Masque模型在BLEU-1指标上提升了8.83%,BLEU-4指标上提升了10.69%，ROUGE-L指标上提升了4.6%。该提升较在MS MARCO(NLG)数据集上更为显著，主要原因为NarrativeQA的答案更偏向于推理性质的概括总结，而MS MARCO(NLG)的答案则更偏向于基于段落中的答案片段进行完整的表述，这也表明了MS MARCO(NLG)的任务难度比NarrativeQA小，预训练模型在推理方法更占优势。

## 5 结语

本文针对生成式阅读理解模型缺乏答案边界和问题分类信息理解的问题，提出一种基于多任务学习的生成式阅读理解模型，通过答案抽取模型和问题分类模型优化生成式阅读理解模型。在三个阅读理解数据集上的实验结果表明，本文提出的基于多任务的生成式阅读理解模型能够有效地学习答案的边界信息和问题分类信息，在三个数据集上均取得了目前生成式模型的最好性能。在未来的工作中，我们将研究如何将该模型迁移至面向长文本的机器阅读理解任务上，使得该模型能够学习整个长文本的同时确定答案的边界信息，并以此生成答案。

## 致谢

本文工作得到国家自然科学基金（基金号61703293，61672368，61672367），江苏高校优势学科建设工程资助项目资助。

## 参考文献

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. MS MARCO: a human-generated machine reading comprehension dataset. *Computing Research Repository (CoRR)*, arXiv:1611.09268. Version 3.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, NanYang, Xiaodong Liu, Yu Wang, Songhao Piao, Jian-feng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. *arXiv preprint arXiv:2002.12804*.

Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. In *Empirical Methods in NaturalLanguage Processing (EMNLP)*, pages 4220–4230.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the EMNLP*, pages 2174– 2184, Brussels, Belgium.

Jacob Devlin, MingWei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *Computing Research Repository (CoRR)*, arXiv:1810.04805. Version 1.

Li Dong, 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 *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Matthew Dunn, Levent Sagun, Mike Higgins, 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*.

Mandar Joshi, 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*.

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

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proc. ACL Workshop on Text Summarization Branches Out*.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *Computing Research Repository (CoRR)*, arXiv:1806.08730. Version 1.

Kyosuke Nishida, Itsumi Saito, Kosuke Nishida, Kazutoshi Shinoda, Atsushi Otsuka, Hisako Asano, and Junji Tomita. 2019. Multi-style generative reading comprehension. In *ACL*.

Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Association for Computational Linguistics (ACL)*, pages 311–318.

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

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. CoQA: A conversational question answering challenge. *Computing Research Repository (CoRR)*, arXiv:1808.07042. Version 1.

Ilya Sutskever, Oriol Vinyals, and Quoc V.Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in neural information processing systems*, pages 3104–3112.

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

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *Proceedings of the 36th InternationalConference on Machine Learning*, volume 97, pages 5926–5936. PMLR.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. In *AAAI*.

Chuanqi Tan, 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 *Association for the Ad-vancement of Artificial Intelligence (AAAI)*, pages 5940–5947.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukaszKaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors,*Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2692–2700.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2018. Reinforced reader-ranker for open-domain question answering. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 5981–5988.

Dongling Xiao, Han Zhang, Yukun Li, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE-GEN: An Enhanced Multi-Flow Pre-training and Fine-tuning Framework for Natural Language Generation. *arXiv preprint arXiv:2001.11314*.

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. In EMNLP.