Multi-Perspective Context Aggregation for Semi-supervised Cloze-style Reading Comprehension

Liang Wang[1], **Sujian Li**[2], **Wei Zhao**[1],
Kewei Shen[1], **Meng Sun**[1], **Ruoyu Jia**[1], **Jingming Liu**[1]
[1]Yuanfudao Research, Beijing, China
[2]Key Laboratory of Computational Linguistics, Peking University, MOE, China
{wangliang01,zhaowei01,shenkw,sunmeng,jiary,liujm}@fenbi.com
lisujian@pku.edu.cn

Abstract

Cloze-style reading comprehension has been a popular task for measuring the progress of natural language understanding in recent years. In this paper, we design a novel multi-perspective framework, which can be seen as the joint training of heterogeneous experts and aggregate context information from different perspectives. Each perspective is modeled by a simple aggregation module. The outputs of multiple aggregation modules are fed into a one-timestep pointer network to get the final answer. At the same time, to tackle the problem of insufficient labeled data, we propose an efficient sampling mechanism to automatically generate more training examples by matching the distribution of candidates between labeled and unlabeled data. We conduct our experiments on a recently released cloze-test dataset CLOTH (Xie et al., 2017), which consists of nearly $100k$ questions designed by professional teachers. Results show that our method achieves new state-of-the-art performance over previous strong baselines.

1 Introduction

Reading comprehension is a challenging task which requires the deep understanding of natural language. Cloze test is a particular form of reading comprehension: given a passage with blanks, an examinee is required to fill in the missing word (or phrase) that best fits the context surrounding the blank. Recently, cloze-style reading comprehension has drawn growing interests from NLP research communities, since such a task meets the practical requirements and is relatively easy to design.

The research of cloze-style reading comprehension is first advanced by two large-scale corpora: the CNN/Daily Mail (Hermann et al., 2015) and CBT (Hill et al., 2015) datasets, which are automatically constructed by randomly or periodically deleting a word from original passage. Though the automatically generated datasets usually consist of a large quantity of labeled data and make it possible to train large neural network models, they are in nature far away from real-world language understanding problems and have serious ambiguity issues (Chen et al., 2016). As a result, the state-of-the-art system of cloze test almost reaches the performance ceiling and loses its improvement direction due to the limitation of the corpus (Chen et al., 2016). In such situation, Xie et al. (2017) argues that it is a more reliable means to assess language proficiency with carefully designed questions by professional teachers, and releases a novel corpus CLOTH. The CLOTH dataset brings the new challenge of exploring a comprehensive evaluation of language proficiency and specifically divides the questions into several types including matching, reasoning and grammar etc. Table 1 shows several example questions from CLOTH.

From experiments by Xie et al. (2017), we can see that the *Stanford attention reader* (Chen et al., 2016) of having the near state-of-the-art performance (with an accuracy of about 0.74) on CNN/Daily Mail only gets an accuracy of 0.487 on CLOTH and there exists a huge performance gap between human and popular machine learning models. The main reason is that attention models are mainly good at processing matching questions (e.g., the first example in Table 1 has matching between *"police"* and *"accident"*, *"man died"*), which occupy a less percentage in CLOTH than in CNN/Daily Mail. Xie

Proceedings of the 27th International Conference on Computational Linguistics, pages 857–867
Santa Fe, New Mexico, USA, August 20-26, 2018.

question	type
..... As a Senior student, I have to __ many exams. ..... *A: finish* *B: win* ***C: take*** *D: join*	collocation
I am calling from the __ station ..... " There was an accident, and a man died ." ..... *A: post* *B: bus* ***C: police*** *D: railway*	matching
a student reported that I made an error ..... He was __ and after thanking him for his honesty ..... he said angrily . *A: wise* ***B: right*** *C: rigid* *D: angry*	reasoning
..... They are used to __ messages by computers and smart phones. ..... ***A: sending*** *B: send* *C: sent* *D: sends*	grammar

Table 1: Example questions and their corresponding types from the CLOTH dataset. "....." represents some omitted irrelevant sentences.

et al. (2017) also present the word-predicting potential of language models (LM) which can well tackle lexical collocation (e.g., *"take"* and *"exam"* in the second example), given a large volume of unlabeled data and high computation power. Furthermore, Xie et al. (2017) points out that the most difficult questions belong to the long-term-reasoning type (e.g., the third example question), which constitutes approximately 22.4% in CLOTH and needs more semantics to deal with.

To comprehensively consider the progress and questions in CLOTH, we come up with the idea of modeling multiple perspectives to arrive at the correct answer, given limited computation power. Our multi-perspective network consists of several parallel modules, where each module aggregates context information from a unique perspective. We model long-distance matching with attentive readers, global semantics with iterative dilated convolutions and lexical collocation with both n-gram and neural language model(LM). The outputs of aggregation modules are further integrated and fed into a one-timestep pointer network (Vinyals et al., 2015) to get the final answer.

Next, one challenging problem is how to effectively train our multi-perspective network due to the insufficiency of labeled data. To overcome this problem, Xie et al. (2017) present a representativeness-based weighted loss function. Their approach has two drawbacks: first, it requires to train another model for predicting a candidate's representativeness score; second, it is not a sample-efficient way since each word including uninformative stop words becomes a training example. In this paper, we improve on Xie et al. (2017)'s approach and develop a semi-supervised learning method by matching the distribution of candidates between labeled and unlabeled data. The intuition is to make automatically constructed data as similar as possible to existing labeled data. Stop words, named entities and out-of-vocabulary words should be downsampled while meaningful content words should be kept for training.

Our method is simple, straightforward and shows better performance with only a fraction of training examples. Experiments show that our semi-supervised multi-perspective network is able to outperform state-of-the-art results on the CLOTH dataset by 4.2%.

2 Model

Formally, the task of cloze-style reading comprehension requires choosing the correct answer from $|c|$ candidates $\{c_i\}_{i=1}^{|c|}$ given a sequence of words $\{w_i\}_{i=1}^{n}$ as context. Candidate c_i could be a word or a phrase. For the CLOTH dataset, each question has $|c| = 4$ candidates.

2.1 MPNet: Multi-Perspective Context Aggregation Network

The overall architecture of our proposed MPNet is shown in Figure 1. It consists of an input layer, a multi-perspective aggregation layer and an output layer.

Input Layer Given the passage as a variable-length word sequence $\{w_i\}_{i=1}^{n}$, we embed each word into 300-dimensional word embeddings $\{\mathbf{e}_i\}_{i=1}^{n}$ using GloVe vectors. Then, we apply bidirectional GRU(BiGRU) on $\{\mathbf{e}_i\}_{i=1}^{n}$ to get contextualized word representations $\{\mathbf{h}_i\}_{i=1}^{n}$ (McCann et al., 2017)

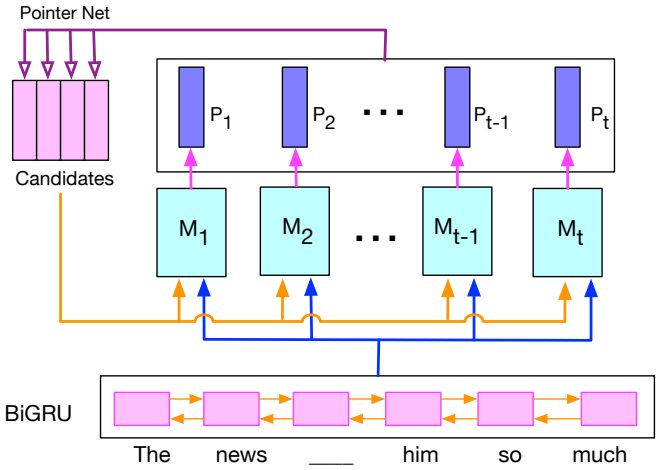

Figure 1: MPNet: **M**ulti-**P**erspective context aggregation network. We only show part of the context *"The news __ him so much"* and "__" is the blank to fill in.

(Peters et al., 2018), since GRU is computationally more efficient and shows slightly better performance than LSTM.

$$\begin{aligned}\overrightarrow{\mathbf{h}}_i &= \overrightarrow{GRU}(\overrightarrow{\mathbf{h}}_{i-1}, \mathbf{e}_i) \\ \overleftarrow{\mathbf{h}}_i &= \overleftarrow{GRU}(\overleftarrow{\mathbf{h}}_{i+1}, \mathbf{e}_i) \\ \mathbf{h}_i &= [\overrightarrow{\mathbf{h}}_i; \overleftarrow{\mathbf{h}}_i]\end{aligned} \tag{1}$$

We also use another GRU to encode candidates $\{c_i\}_{i=1}^{|c|}$ into fixed-length vectors $\{\mathbf{u}_i\}_{i=1}^{|c|}$, as candidates may be multi-word phrases.

Multi-Perspective Aggregation Layer This layer consists of several independent aggregation modules $\{M_i\}_{i=1}^{t}$. Computation can be easily parallelized since modules are independent. Each module M_i takes contextualized word representations $\{\mathbf{h}_i\}_{i=1}^{n}$ and candidates' encoding $\{\mathbf{u}_i\}_{i=1}^{|c|}$ as input and outputs a vector $\mathbf{p}_i$, which reflects the information from module M_i's perspective. We also assume aggregation modules can have access to $\{w_i\}_{i=1}^{n}$ and $\{c_i\}_{i=1}^{|c|}$. For cloze-style reading comprehension, each module should be able to distill some knowledge which can judge whether a candidate fits a given context from a perspective.

The aggregation modules that we use are listed below:

- **Selective Copying** Assuming the index of the blank is j, this module simply selects the hidden representation of the blank $\mathbf{h}_j$, directly copies it to the output $\mathbf{p}_{sc}$ and ignores everything else. Note that $\mathbf{h}_j$ is the output of BiGRU and already incorporates context information from both forward and backward directions. This resembles a bidirectional language model without softmax output layer. Words near the blank are paid more attention which is consistent with our intuition of filling in the blank.

- **Attentive Reader** A large portion of questions involve matching candidates with related words which may be far away from each other such as the second example in Table 1. *Attentive reader* proposed by Chen et al. (2016) directly attends to the entire context and therefore avoids the difficulty of modeling long-range dependence. Original bilinear attention function (Chen et al., 2016) is slightly modified by introducing $\mathbf{b}_{ar}$ to model attention bias towards the ith word. $\mathbf{u}$ is the vector representation of a candidate, we omit its subscript for simplicity.

$$\begin{aligned}\alpha_i &= \text{softmax}_i(\mathbf{u}^T\mathbf{W}_{ar}\mathbf{h}_i + \mathbf{b}_{ar}^T\mathbf{h}_i), i \in [1, n] \\ \mathbf{p}_{ar} &= \sum_{i=1}^{n} \alpha_i \mathbf{h}_i\end{aligned} \tag{2}$$

- **Iterative Dilated Convolution** Convolutional neural networks have been a successful method for modeling both natural language (Kim, 2014) and images. Multiple layers of CNNs can extract features in a hierarchical way, which shares similarity with the compositional property of natural language. Dilated convolution is a variant of traditional convolution and is more efficient for multi-scale context aggregation (Yu and Koltun, 2015; Strubell et al., 2017). In this work, we use two blocks where each block consists of two dilated convolutions with dilation rate set to 1 and 3 respectively. Max pooling across filters is applied to get the final output $\mathbf{p}_{idc}$.

- **N-gram Statistics** To explicitly incorporate collocation information, we use this module to output logarithmic n-gram counts $\mathbf{p}_{ng}$ from *English Wikipedia* with $n \in [1, 5]$. Logarithmic function avoids the optimization difficulty with extremely large numbers.

Note that the output $\mathbf{p}_{sc}$ from selective copying module and $\mathbf{p}_{idc}$ from iterative dilated convolution module don't depend on the candidates. We therefore get context representation $\mathbf{P}_{ctx} = [\mathbf{p}_{sc}; \mathbf{p}_{idc}]$ by concatenating $\mathbf{p}_{sc}$ and $\mathbf{p}_{idc}$. Similarly, we can get the representation vector $\mathbf{C}_i$ for ith candidate by concatenating the output $\mathbf{p}^i_{ar}$ from attentive reader module, $\mathbf{p}^i_{ng}$ from n-gram statistics and $\mathbf{u}_i$ from the candidate encoder: $\mathbf{C}_i = [\mathbf{u}_i; \mathbf{p}^i_{ar}; \mathbf{p}^i_{ng}], i \in [1, |c|]$.

Output Layer We use a one-timestep pointer network (Vinyals et al., 2015) to choose the correct answer from $|c|$ candidates $\{\mathbf{c}_i\}_{i=1}^{|c|}$. Given context representation $\mathbf{P}_{ctx}$ and candidates representation $\{\mathbf{C}_i\}_{i=1}^{|c|}$, we first refine candidates representation with a gating mechanism:

$$\begin{aligned} \mathbf{g}_i &= \sigma(\mathbf{W}_1\mathbf{P}_{ctx} + \mathbf{W}_2\mathbf{C}_i + \mathbf{b}),\ i \in [1, |c|] \\ \mathbf{C'}_i &= \mathbf{C}_i \odot \mathbf{g}_i,\ i \in [1, |c|] \end{aligned} \tag{3}$$

σ is the sigmoid function and $\odot$ denotes pointwise multiplication. Then we calculate the distribution of being the correct answer over candidates with bilinear function:

$$\hat{y}_i = \mathrm{softmax}_i(\mathbf{C'}_i^T \mathbf{W}_o \mathbf{P}_{ctx} + \mathbf{b}_o^T \mathbf{C'}_i),\ i \in [1, |c|] \tag{4}$$

$\{\hat{y}_i\}_{i=1}^{|c|}$ is a probability distribution and the pointer points to the candidate $\arg\max_i(\hat{y}_i)$.

Model Learning The model is trained by minimizing the standard cross-entropy loss.

Discussion Different aggregation modules summarize context from different perspectives. In order to precisely locate the correct answer, a set of complementary aggregation modules are preferred where one module may only focus on lexical collocation and another module may be sensitive to the global matching. It is worth noting that our MPNet framework can be easily extended by adding other effective aggregation modules.

In addition, the main idea of MPNet is to some extent connected with the mixture of experts (MoE) (Masoudnia and Ebrahimpour, 2014). If each aggregation module can be seen as an expert, then multiple aggregation modules become MoE. One key difference is that aggregation modules in MPNet have heterogeneous network structures while traditional MoE models usually consist of homogeneous experts.

2.2 SemiMPNet: Semi-supervised Learning with Distribution Matching

SemiMPNet is the semi-supervised variant of our proposed MPNet in Section 2.1, with exactly the same network architecture. Though CLOTH consists of nearly $100k$ questions, it is generally not enough to train large neural models. Semi-supervised learning comes to the rescue. We propose to sample from unlabeled text to construct training examples automatically. In order to train effectively, we need to make the automatically generated data similar to labeled data and ensure that candidates should have a similar distribution in original labeled data to that in the generated data. Then, we formulate candidates distribution matching in two datasets as two sampling problems as follows:

How to sample positive candidates? We assume D_u is a collection of unlabeled documents, D_c is the collection of all candidates in the CLOTH dataset and V is the vocabulary which is composed of all the candidates occurring in CLOTH. Each word $w_i \in V$ is associated with an unknown sampling probability $p(w_i)$. To match the distribution of candidates between D_u and D_c, the following constraints about $p(w_i), i \in [1, |V|]$ should be satisfied:

$$\begin{gathered} \frac{p(w_i)\#(w_i, D_u)}{\sum_{j=1}^{|V|} p(w_j)\#(w_j, D_u)} = \frac{\#(w_i, D_c)}{\sum_{j=1}^{|V|} \#(w_j, D_c)},\ i \in [1, |V|] \\ 0 \leq p(w_i) \leq 1, i \in [1, |V|] \\ \max\{p(w_i)|\ i \in [1, |V|]\} = 1 \end{gathered} \tag{5}$$

Function $\#(w_i, D)$ returns the frequency of w_i in corpus D. The second constraint is to make sure $\{p(w_i)\}_{i=1}^{|V|}$ is a valid probability distribution and the third constraint is to make full use of data. There is generally no exact solution to Equation(5) as $\#(w_i, D_u) = 0$ and $\#(w_i, D_c) > 0$ may hold for some i. Instead, we use an approximate solution:

$$p(w_i) = \min(1, \frac{\#(w_i, D_c)}{\#(w_i, D_u)} \times \frac{\gamma \sum_{j=1}^{|V|} \#(w_j, D_u)}{\sum_{j=1}^{|V|} \#(w_j, D_c)}) \tag{6}$$

The coefficient γ can be interpreted as the average probability of sampling a word. We set $\gamma = 0.5$ based on validation data. With this strategy, we sample the positive candidates and use the corresponding passages as their contexts.

How to sample negative candidates? Given a positive candidate w_p, the probability of w_i being sampled as a negative candidate $p(w_i|w_p)$ can be calculated as follows:

$$p(w_i|w_p) = \frac{\lambda}{|V|} + (1 - \lambda)\frac{\#(w_i, w_p)}{\sum_{j=1}^{|V|} \#(w_j, w_p)} \tag{7}$$

$\#(w_i, w_p)$ is the co-occurrence counts of w_i and w_p as candidates in labeled dataset D_c. Intuitively, the co-occurrence probability of w_i and w_p should match between constructed data and labeled data. λ is the probability of randomly selecting a word from the entire vocabulary, similar to the exploration mechanism in reinforcement learning. It makes our model more robust to overfitting and we set $\lambda = 0.1$ throughout the experiments.

In the case that candidates are multi-word phrases, our method is also applicable by simply expanding the vocabulary V to include phrases in D_c.

3 Experiments

3.1 Experimental Setup

Dataset and Evaluation Metrics We use the CLOTH (Xie et al., 2017) dataset for training and evaluation. RACE (Lai et al., 2017) dataset and *English Wikipedia* [1] serve as background text corpora for semi-supervised learning. RACE dataset consists of nearly $28k$ reading comprehension passages from high-school examinations. We delete passages that have a Jaccard similarity over 0.85 with passages in the CLOTH dataset. Furthermore, background text corpora also include training passages from the CLOTH dataset by filling the correct answer back into the corresponding blank.

Accuracy is used as the evaluation metric. To make a fair comparison with Xie et al. (2017), we also report performance on CLOTH-M(middle school questions) and CLOTH-H(high school questions).

Hyperparameters Our model is implemented with Tensorflow (Abadi et al., 2016). Hyperparameters are optimized with random search based on validation data. All our models are run on a single GPU(Tesla P40). NLTK (Bird and Loper, 2004) is used for tokenization. Word embeddings are initialized with

[1] https://dumps.wikimedia.org/enwiki/

300-dimensional GloVe (Pennington et al., 2014) vectors. Only vectors of top 1000 frequent words are fine-tuned during training. Our network is trained with Adam algorithm (Kingma and Ba, 2014). The initial learning rate is set to 10^{-3}. We decrease learning rate to 10^{-4} after $15k$ iterations and further decrease it to 10^{-5} after $50k$ iterations. Both forward and backward GRU have 128 hidden units. For input, we use a context window of 80 words. For 1D dilated convolution, we use 2 blocks, the number of filters is 128 and the convolution width is 3 for all layers. Batch normalization and ReLU are applied on top of convolution. Gradients are clipped to have a maximum L2 norm of 5. Dropout with probability 0.5 is applied to the output of BiGRU.

Model	+ constructed data?	CLOTH	CLOTH-M	CLOTH-H
Random	No	25.0%	25.0%	25.0%
LSTM (Xie et al., 2017)	No	48.4%	51.8%	47.1%
Stanford Attention Reader (Chen et al., 2016)	No	48.7%	52.9%	47.1%
MPNet - ngram	No	**50.1%**	**53.2%**	**49.0%**
Language Model (Xie et al., 2017)	Yes	54.8%	64.6%	50.6%
Representativeness (Xie et al., 2017)	Yes	56.5%	66.5%	52.6%
LSTM + Representativeness (Xie et al., 2017)	Yes	58.3%	67.3%	54.9%
SemiMPNet - ngram	Yes	**60.9%**	**67.6%**	**58.3%**
Human	–	86.0%	89.7%	84.5%

Table 2: Experimental results without using external data. We exclude *n-gram* as *n-gram* is calculated based on external corpus *Wikipedia*. SemiMPNet uses passages from CLOTH for semi-supervised data augmentation. Human performance is from Xie et al. (2017).

3.2 Baselines

LSTM is a baseline model by Xie et al. (2017). First, a BiLSTM layer is applied to context word embeddings. Then it uses the outputs near the blank to calculate the probability of being the correct answer for each candidate.

Stanford Attention Reader is an attention-based neural model for reading comprehension presented by Chen et al. (2016). Experimental results are from Xie et al. (2017).

Language Model To overcome the difficulty of insufficient labeled data. Xie et al. (2017) propose to train a neural language model on passages from the CLOTH dataset. The candidate that results in the highest probability is chosen as the predicted answer. It's fair to say *Language Model* is a simple data augmentation approach that treats every word as a training example with equal weight.

Representativeness is another semi-supervised data augmentation approach by Xie et al. (2017). It assigns different weights to different constructed examples based on *Representativeness score*. *Representativeness* can be interpreted as the probability of a given word being selected as a blank by human. For more technical details, please refer to Xie et al. (2017).

One-billion-word-LM is a state-of-the-art neural language model (Jozefowicz et al., 2016) trained on one-billion-word benchmark (Chelba et al., 2013). It has more than 1 billion parameters and is publicly available[2].

3.3 Main Results

We evaluate our model's performance in two experimental settings: use external data or not. For the setting without external data, we only use passages from CLOTH for training and semi-supervised data augmentation. Though GloVe vectors are trained on external text corpora, it has become a standard practice for NLP to use pretrained embeddings. Therefore, GloVe vectors are used in both settings and so does the work by Xie et al. (2017).

Results w/o External Data Results are shown in Table 2. When trained only on labeled data, both LSTM by Xie et al. (2017) and our proposed MPNet perform poorly, though MPNet slightly outper-

[2] https://github.com/tensorflow/models/tree/master/research/lm_1b

forms LSTM by 1.7% in overall accuracy. The accuracy of middle school questions (CLOTH-M) is consistently higher than high school questions (CLOTH-H) across all of our experiments, since middle school questions are relatively easier.

w	$p(w)$	w	$p(w)$	w	$p(w)$
I	0.04	festivals	0.75	California	0.13
the	0.03	birthday	1.0	thank you	0.26
Frank	0.09	8	0.09	congratulation	1.0

Table 3: Sample probability for some words. $p(w)$ is the probability of sampling w to construct a training example. Stop words, named entities usually have low probability. See Section 2.2 for details.

Table 2 clearly shows that constructed data can significantly boost both models' performance. Xie et al. (2017) explore several different ways for data augmentation: *Language Model* treats every word equally, while *Representativeness* method assigns different weights to different words by training an representativeness prediction network. This mechanism improves the accuracy from 48.4% to 58.3%. Further, our proposed method adopts a new sampling method and requires sampling words with distribution constraints, which makes training more efficient. As shown in Table 3, stop words (e.g., *"I"* and *"the"*), named entities (e.g., *"Frank"* and *"California"*) and common phrases (e.g., *"thank you"*) have low probability of being sampled. Content words such as *"festivals"* and *"birthday"* are more likely to be sampled. One limitation of our sampling method is its inability to handle synonyms. Since synonyms tend to co-occur as candidates in the labeled dataset, this problem is not as severe as it looks like to be. *"SemiMPNet - ngram"* beats all baseline methods and achieves the highest accuracy 60.9%. Human performance is 86.0% which is much higher than *"SemiMPNet - ngram"*. The effectiveness of constructed data indicates that the lack of labeled data has become a bottleneck.

Model	CLOTH	CLOTH-M	CLOTH-H
One-billion-word-LM	70.7%	74.5%	69.3%
MPNet	65.3%	70.0%	63.6%
SemiMPNet	70.4%	75.5%	68.5%
SemiMPNet + One-billion-word-LM	**74.9%**	**79.0%**	**73.3%**
Human	86.0%	89.7%	84.5%

Table 4: Experimental results with external data. SemiMPNet use passages from the RACE dataset for semi-supervised data augmentation.

Results with External Data As shown in Table 4, incorporation of the RACE dataset for semi-supervised learning improves accuracy from 65.3% to 70.4%. However, MPNet and SemiMPNet still underperform a pretrained state-of-the-art neural language model *One-billion-word-LM* (Jozefowicz et al., 2016). It is trained on a large corpus with nearly 1 billion words and achieves an accuracy of 70.7%. In contrast, SemiMPNet is trained on only $\sim$ 10 million words and has a 0.3% gap in accuracy, which is pretty impressive given that the sizes of two corpora differ by two orders of magnitude. Once again it shows the power of transferring knowledge from unlabeled text corpora.

As a further discussion, we'd like to point out that although language model can achieve good results, it is not the most efficient way. Actually, experimental results in Table 2 show that language model underperforms SemiMPNet given the same amount of text. A fair comparison would be training SemiMPNet on the one-billion-word benchmark. Considering the size of the one-billion-word corpus, applying our semi-supervised method directly on the one-billion-word corpus would require a sizable amount of computing power. Here we design an approximate method *"SemiMPNet + One-billion-word-LM"* and combine MPNet and *One-billion-word-LM* by linear interpolation of their output probabilities:

$$p = \beta p_{mp} + (1 - \beta) p_{lm} \tag{8}$$

p_{mp} is the output probability by SmiMPNet and p_{lm} is the normalized probability by *One-billion-word-*

LM[3]. Setting $\beta = 0.5$ yields empirically good results. Hyper-parameter search shows the results are quite robust to a wide range of β values. We can see that our model *"SemiMPNet + One-billion-word-LM"* achieves a new state-of-the-art performance of 74.9%, which improves *One-billion-word-LM* by 4.2%. This also shows the complementarity of SemiMPNet and *One-billion-word-LM*. Two models can learn different aspects of the contexts.

3.4 Ablation Study

Our proposed MPNet consists of four aggregation modules. To examine the effect of each module, we conduct an ablation study. The results are shown in Table 5.

Model	CLOTH
SemiMPNet	**70.4%**
w/o selective copying	69.4% (-1.0)
w/o attentive reader	67.6% (-2.8)
w/o dilated convolution	69.6% (-0.8)
w/o n-gram statistics	63.0% (-7.4)

Table 5: Results for SemiMPNet ablation study. RACE dataset is used for semi-supervised data augmentation.

N-gram statistics turn out to be the single most influential factor. Overall performance decreases by 7.4% without *n-gram statistics*. On one hand, this result further highlights the importance of distilling knowledge from large text corpora. On the other hand, it proves that our background corpus is not large enough for neural models to learn reliable lexical collocation information.

Attentive reader also has a significant impact on overall performance. Attention mechanism is able to locate useful information regardless of its positional distance from the blank. In contrast, RNNs need to preserve such information over a long distance which is nontrivial.

Besides, the results in Table 5 support an important intuition in this paper: different modules capture context information from different perspectives, and removing any one of them would result in decreased performance.

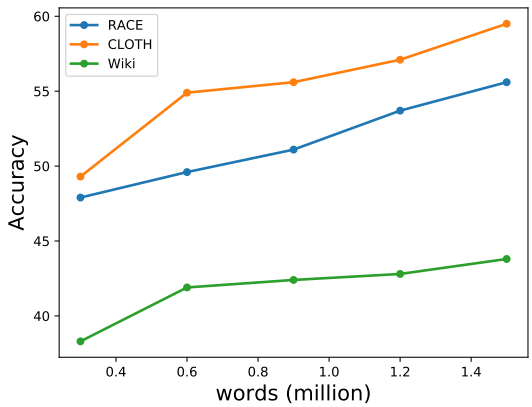

Figure 2: Examining effects of different text corpora. The x-axis is the number of words in background corpus, and the y-axis is the accuracy on test data. To avoid the influence of external data, we report model performance with *"SemiMPNet - ngram"*.

3.5 Examining Effects of Background Corpus

For our semi-supervised learning model SemiMPNet, background corpus is used to construct training examples. The choice of background corpus can make a big difference. In this section, we conduct an

[3]The normalization makes sure the probabilities for all candidates sum to 1.

experiment to examine such effects. Three different corpora are used: passages from the training set of CLOTH, RACE and English Wikipedia.

Results are shown in Figure 2. Unsurprisingly, more data lead to better performance. Moreover, given the same amount of text, CLOTH consistently beats RACE and RACE consistently beats Wikipedia. As we know, CLOTH and RACE consists of passages designed for high school students, while Wikipedia entries are for the general public and therefore have a different word distribution. Thus, how to make use of huge unlabeled data to help training is a key for performance improvement, since training corpora of higher quality are generally smaller in scale.

4 Related Work

Reading Comprension or machine reading is drawing more and more interests among NLP research communities. The CNN/Daily Mail (Hermann et al., 2015) and CBT (Hill et al., 2015) are two automatically generated cloze-style datasets. Though they can be large in scale, the quality of automatically generated questions is generally lower than manually labeled ones. Instead, SQuAD (Rajpurkar et al., 2016) adopts a crowd-sourcing approach to ensure its quality. In SQuAD, each passage accompanies one or more questions and the answer is a text span of the given passage for the convenience of automatic evaluation. Rapid progress has been made with neural network based models (Wang et al., 2016). The performances of state-of-the-art models on SQuAD such as QANet (Yu et al., 2018) and ELMo (Peters et al., 2018) are already very close to human. There are also some datasets focusing on answering questions from real-world scenarios. MS MARCO (Nguyen et al., 2016) and DuReader (He et al., 2017) are two typical examples. Such datasets are usually harder as they require the ability of both comprehension and language generation. BLEU and ROUGE are often used as evaluation metrics. One potential problem is that answers with high BLEU scores may have very different semantic meanings.

Cloze Test is a particular form of reading comprehension task and has been widely adopted as a method for assessing students' language proficiency. Zweig and Burges (2011) presented a challenging dataset for sentence completion but its scale is too small with only 1040 questions. CNN/Daily Mail (Hermann et al., 2015), CBT (Hill et al., 2015), LAMBADA (Paperno et al., 2016) and CLOTH (Xie et al., 2017) are all large-scale cloze-test datasets, with the difference that each question in CLOTH has four candidate options. Recently proposed Story Cloze (Mostafazadeh et al., 2017) is a cloze-test dataset that goes beyond words and phrases, and requires choosing a sentence as the appropriate story ending.

Semi-supervised Methods for reading comprehension are widely studied due to the fact that labeled data is scarce. One major approach is pretraining a model for text representation and reusing the weights during supervised learning. Autoencoders (Hewlett et al., 2017), machine translation (McCann et al., 2017) and language model (Peters et al., 2018) can be used for representation learning. Another approach aims to directly construct training examples from unlabeled text corpora. Weighted loss function (Xie et al., 2017) and reinforcement learning (Yang et al., 2017) can be used to alleviate the discrepancy between human-labeled data and automatically-constructed data.

5 Conclusion

In this paper, we propose a multi-perspective network MPNet for cloze-style reading comprehension. MPNet consists of several parallel context aggregation modules. Each module summarizes the variable-length context and candidates into a fixed-length vector from a unique perspective. We explore four effective implementations of aggregation modules in experiments. The architecture of MPNet is very flexible and can be easily extended by adding more task-specific modules.

To overcome the difficulty of limited labeled data, we turn to semi-supervised learning by automatically constructing training examples from unlabeled text corpora. Experiments on the CLOTH dataset show that our semi-supervised MPNet achieves new state-of-the-art performance. In our future work, we'd like to come up with more effective methods to tackle this challenge.

Acknowledgements

We would like to thank three anonymous reviewers for their insightful comments, and COLING 2018 organizers for their efforts.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. TensorFlow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283.

Steven Bird and Edward Loper. 2004. NLTK: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 31. Association for Computational Linguistics.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

Danqi Chen, 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)*, volume 1, pages 2358–2367.

Wei He, Kai Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, Xuan Liu, et al. 2017. DuReader: a Chinese Machine Reading Comprehension Dataset from Real-world Applications. *arXiv preprint arXiv:1711.05073*.

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*, pages 1693–1701.

Daniel Hewlett, Llion Jones, Alexandre Lacoste, et al. 2017. Accurate supervised and semi-supervised machine reading for long documents. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2001–2010.

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

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Saeed Masoudnia and Reza Ebrahimpour. 2014. Mixture of experts: a literature survey. *Artificial Intelligence Review*, pages 1–19.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. *arXiv preprint arXiv:1708.00107*.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James F Allen. 2017. LSDSem 2017 Shared Task: The Story Cloze Test. *LSDSem 2017*, page 46.

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*.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*.

Jeffrey Pennington, Richard Socher, and Christopher 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.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *NAACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Emma Strubell, Patrick Verga, David Belanger, and Andrew McCallum. 2017. Fast and accurate entity recognition with iterated dilated convolutions. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2660–2670.

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

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

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

Zhilin Yang, Junjie Hu, Ruslan Salakhutdinov, and William W Cohen. 2017. Semi-Supervised QA with Generative Domain-Adaptive Nets. *arXiv preprint arXiv:1702.02206*.

Fisher Yu and Vladlen Koltun. 2015. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

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

Geoffrey Zweig and Christopher JC Burges. 2011. The Microsoft Research sentence completion challenge. Technical report, Technical Report MSR-TR-2011-129, Microsoft.