DRr-Net: Dynamic Re-Read Network for Sentence Semantic Matching

Kun Zhang,[1] **Guangyi Lv,**[1] **Linyuan Wang,**[1] **Le Wu,**[2] **Enhong Chen,**[1,*] **Fangzhao Wu,**[3] **Xing Xie**[3]
[1]Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China
{zhkun, gylv, wly757}@mail.ustc.edu.cn, cheneh@ustc.edu.cn
[2]Hefei University of Technology, China, lewu@hfut.edu.cn
[3] Microsoft Research Asia, China, wufangzhao@gmail.com, xing.xie@microsoft.com

Abstract

Sentence semantic matching requires an agent to determine the semantic relation between two sentences, which is widely used in various natural language tasks such as Natural Language Inference (NLI) and Paraphrase Identification (PI). Among all matching methods, attention mechanism plays an important role in capturing the semantic relations and properly aligning the elements of two sentences. Previous methods utilized attention mechanism to select important parts of sentences at one time. However, the important parts of the sentence during semantic matching are dynamically changing with the degree of sentence understanding. Selecting the important parts at one time may be insufficient for semantic understanding. To this end, we propose a *Dynamic Re-read Network (DRr-Net)* approach for sentence semantic matching, which is able to pay close attention to a small region of sentences at each step and re-read the important words for better sentence semantic understanding. To be specific, we first employ Attention Stack-GRU (ASG) unit to model the original sentence repeatedly and preserve all the information from bottom-most word embedding input to up-most recurrent output. Second, we utilize Dynamic Re-read (DRr) unit to pay close attention to one important word at one time with the consideration of learned information and re-read the important words for better sentence semantic understanding. Extensive experiments on three sentence matching benchmark datasets demonstrate that *DRr-Net* has the ability to model sentence semantic more precisely and significantly improve the performance of sentence semantic matching. In addition, it is very interesting that some of finding in our experiments are consistent with the findings of psychological research.

1 Introduction

Sentence semantic matching, a fundamental technology in natural language processing, requires an agent to predict the semantic relation between two sentences. For example, in Natural Language Inference (NLI), sentence semantic matching is utilized to determine whether a hypothesis sentence can reasonably be inferred from a given premise sentence (Kim et al. 2018). In Paraphrase Identification (PI), it is utilized to identify whether two sentences express the same meaning or not (Dolan and Brockett 2005).

As a fundamental technology, sentence semantic matching has broad applications, e.g. information retrieval (Clark et al. 2016), question answering (Wang et al. 2017; Liu et al. 2018), and dialog system (Serban et al. 2016). With the large annotated datasets (Bowman et al. 2015; Iyer, Dandekar, and Csernai 2017) and advancement of representation learning techniques (Cheng, Dong, and Lapata 2016; Vaswani et al. 2017), rapid development on sentence semantic matching has been enabled. Among the core semantic matching techniques, attention mechanism plays an important role, which is known for its alignment between representations and modeling the dependency regardless of sequence length. For example, self-attention (Vaswani et al. 2017) can generate better representations by relating elements at different positions in a single sentence. Co-attention (Kim et al. 2018) is capable of modeling sentence interaction in a detailed perspective. They together have become essential for tackling numerous complicated tasks.

However, most of the existing methods select all the important parts of sentences at one time. In fact, the important parts of sentences during semantic matching are dynamically changing with the progress of sentence understanding and should be repeatedly read and processed. For example, when judging the relation between "*a person with a purple shirt is painting an image of a woman on a white wall*" and "*a woman paints a portrait of her best friend*", the important words will change from "*person, purple, shirt, painting, image, woman*" to "*person, image, woman*" in the first sentence, and from "*woman, paints, portrait, best friend*" to "*woman, portrait, best friend*"in the second sentence. As the Chinese proverb says: "*The gist of an article will come to you after reading it over 100 times* ". The important words should be repeatedly read and thought for the final decision.

Moreover, psychological researches have shown that humans only pay attention to a small region of information at one time, i.e., people only focus on 1.5 words each time when intensively reading a piece of text (Wang et al. 1999). Koch and Tsuchiya (2007) have demonstrated that people might focus on less than 7 different objects at the same time. All of the studies indicate that the important words in sentences should be dynamic re-read and re-thought for better semantic understanding and matching.

Inspired by these observations, in this paper, we propose a novel *Dynamic Re-read Network (DRr-Net)* approach for

better sentence semantic matching. To be specific, we utilize Attention Stack-GRU (ASG) unit to model the sentence semantic comprehensively and preserve all the information from bottom-most word embedding input to up-most recurrent output. Then, we use Dynamic Re-read (DRr) unit to pay close attention to one important word at each step and repeatedly read the important words for better sentence semantic understanding. In this way, *DRr-Net* can select the most important word to process with the consideration of learned information, which is in favor of tackling sentence semantic matching task. Extensive evaluations on three sentence matching datasets demonstrate the effectiveness of *DRr-Net* in sentence semantic matching and its advantages over state-of-the-art sentence encoding-based baselines.

2 Related Work

In this section, we will introduce the related works on sentence semantic matching and human attention.

2.1 Sentence Semantic Matching

Sentence semantic matching has achieved a big progress with the development of large annotated data, such as SNLI (Bowman et al. 2015), and Quora Question Pair (Iyer, Dandekar, and Csernai 2017), and various neural network architectures, such as LSTM (Cheng, Dong, and Lapata 2016), GRU (Chung et al. 2014) and attention mechanism (Vaswani et al. 2017; Kun et al. 2018). Among all these methods, attention mechanism plays an important role, which helps models capture the semantic relations and properly align the elements of two sentences. For example, Liu et al. (2016) proposed inner-attention to imitate the human behaviour that concerned more about the important words when reading. Then, they utilized mean pooling to generate the sentence vectors for sentence semantic matching. Shen et al. (2017) developed a directional and multidimensional attention without RNN/CNN structure. They calculated the attention on each dimension of word representations. Then, they utilized a multi-dimensional attention to compress the sequence into a vector, followed by a classification model to compute the final prediction. Kim et al. (2018) utilized densely-connected co-attention network to retain as many features as possible for better sentence understanding. Im and Cho (2017) adopted the masked multi-head attention with distance to explore the sentence semantic. Then, they utilized densely-connected operation to preserve all the information for better sentence semantic matching. However, most of these methods selected all the important parts of sentences at one time. In fact, the important words of sentences are dynamically changing with the progress of sentence understanding. Their methods may be insufficient for better sentence semantic matching.

2.2 Human Attention

Attention mechanism has been widely applied to both natural language process and computer vision domains (Gong, Luo, and Zhang 2017). It allows the model to focus on distinct aspects of the input and thus improve its ability to extract the most relevant parts for outputs (Cho, Courville, and Bengio 2015). Though attention mechanism is helpful for better performance, there is still much to learn in the form of human attention. Psychologists have done plenty of research on this domain. By building an eye tracker, O'Shea (1908) found that some words in a sentence were not fixated when people were reading. Yarbus (1967) described human attention "*is dependent on not only what is shown on the picture, but also the problem facing the observer and the information that he hopes to gain*". Moreover, Wang et al. (1999) found that people may focus on 1.5 words each time when intensively reading a piece of text. Further research (Koch and Tsuchiya 2007; Tononi 2008) demonstrated that people might focus on less than 7 different object at the same time, which meant that human only focused on a small part of information at one time and repeatedly processed important parts for better understanding.

These psychological studies inspired us to utilize the attention mechanism to focus the model on the most important part at each time and repeatedly select the important word with the consideration of learned information for better sentence semantic matching.

3 Problem Statement and Model Structure

In this section, we formulate the sentence semantic matching task as a supervised classification problem and introduce the structure and technical details of *Dynamic Re-read Network (DRr-Net)* for sentence semantic matching.

3.1 Problem Statement

First, we define our task in a formal way. Given two sentence $\boldsymbol{s}^a = \{\boldsymbol{w}_1^a, \boldsymbol{w}_2^a, ..., \boldsymbol{w}_{l_a}^a\}$ and $\boldsymbol{s}^b = \{\boldsymbol{w}_1^b, \boldsymbol{w}_2^b, ..., \boldsymbol{w}_{l_b}^b\}$. Our goal is to learn a classifier ξ which is able to precisely predict the relation $y = \xi(\boldsymbol{s}^a, \boldsymbol{s}^b)$ between $\boldsymbol{s}^a$ and $\boldsymbol{s}^b$. Here, $\boldsymbol{w}_i^a$ and $\boldsymbol{w}_j^b$ are one-hot vectors which represent the i^{th} and j^{th} word in the sentences, and l_a and l_b indicate the total number of words in $\boldsymbol{s}^a$ and $\boldsymbol{s}^b$.

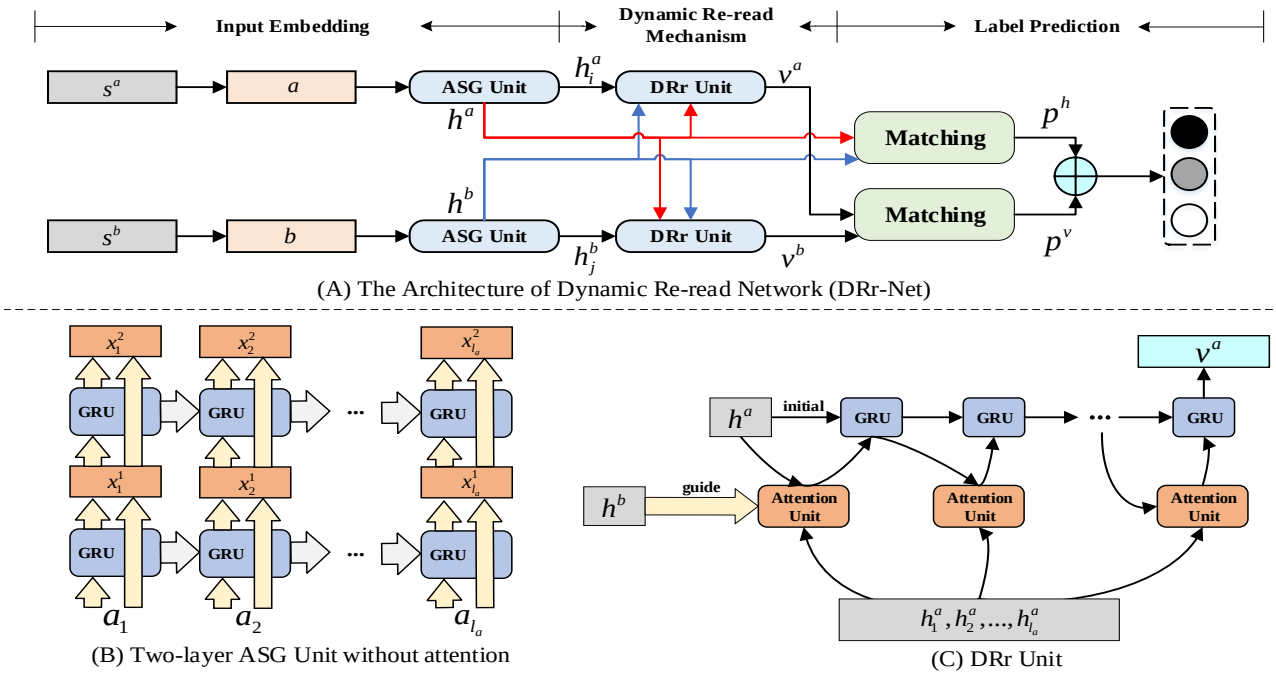
In order to understand sentence semantic more precisely and do better semantic matching, the following important challenge should be considered:

- With the degree of sentence understanding, the important words in the sentence that should be concerned are dynamically changing at each step. How to ensure which word should be paid more attention to at each step with the consideration of learned information?

To this end, we propose a *Dynamic Re-read Network (DRr-Net)* approach for sentence semantic matching.

3.2 Overall Architecture of *DRr-Net*

The overall architecture is shown in Figure 1 (A), which can be classified into three components: 1) *Input Embedding*: encoding each word in the sentence with sufficient features and encoding sentence semantic in a comprehensive way; 2) *Dynamic Re-read Mechanism*: focusing on one important word at each step and repeatedly reading the important words with the consideration of learned information regardless of the words sequence; 3) *Label Prediction*: utilizing original representations and dynamic representations to predict the sentence semantic classification results robustly.



(A) The Architecture of Dynamic Re-read Network (DRr-Net)

(B) Two-layer ASG Unit without attention

(C) DRr Unit

Figure 1: Architecture of *Dynamic Re-read Network (DRr-Net)*.

Input Embedding. This component consists of two parts: word embedding and Attention Stack-GRU (ASG) unit.

1) Word Embedding: The inputs of *DRr-Net* are one-hot representations $\boldsymbol{s}^a = \{\boldsymbol{w}_1^a, \boldsymbol{w}_2^a, ..., \boldsymbol{w}_{l_a}^a\}$ and $\boldsymbol{s}^b = \{\boldsymbol{w}_1^b, \boldsymbol{w}_2^b, ..., \boldsymbol{w}_{l_b}^b\}$ for sentence a and b. For a more comprehensive access to the semantic of each word in sentences, we utilize the concatenation of pre-trained word embedding (Pennington, Socher, and Manning 2014), character features (Gong, Luo, and Zhang 2017), and syntactical features (Chen et al. 2017a; Gururangan et al. 2018) to represent each word in sentences. The character features are obtained by applying a convolutional neural network with a max pooling layer to the learned character embeddings, which can represent words in a finer-granularity and help to avoid the Out-Of-Vocabulary (OOV) problem that pre-trained word vectors suffer from. The syntactical features consist of the embedding of part-of-speech tagging feature, binary exact match feature, and binary antonym feature, which have been proved useful for sentence semantic understanding (Chen et al. 2017a; Gururangan et al. 2018). Next, we pass these representations through a two-layer highway network (Srivastava, Greff, and Schmidhuber 2015) and get the extravagant representations $\{\boldsymbol{a}_i | i = 1, 2, ..., l_a\}$ and $\{\boldsymbol{b}_j | j = 1, 2, ..., l_b\}$ for the words in sentences a and b.

2) Attention Stack-GRU (ASG) Unit. It is beneficial repeatedly reading a sentence multiple times for sentence semantic understanding. In order to model sentence semantic more comprehensively in this way, we employ the stacked Recurrent Neural Network (stack-RNN) (Nie and Bansal 2017), which are composed of multiple RNN layers on the top of each other, to process the sentences. Note that we utilize Gated Recurrent Unit (GRU) as the base unit in *DRr-Net*. To be specific, let $\boldsymbol{H}_l$ be the l^{th} GRU layer. At the time step t, an ordinary stacked RNN is expressed as follows:

$$\boldsymbol{h}_t^l = \boldsymbol{H}_l(\boldsymbol{x}_t^l, h_{t-1}^l), \quad \boldsymbol{x}_t^l = \boldsymbol{h}_t^{l-1}, \tag{1}$$

where $\boldsymbol{x}_t^l$ is the input of the t^{th} step in the l^{th} GRU layer. While this architecture enables us to build up deeper representations, it cannot preserve all the learned information. Even worse this architecture might cause the exploding or vanishing gradient problems. Motivated by (Kim et al. 2018), we concatenate the inputs $x^{(l-1)}$ and the states $h^{(l-1)}$ of the $(l-1)^{th}$ GRU layer as the inputs of the l^{th} GRU layer. In other words, Eq. (1) will be modified as follows:

$$\boldsymbol{h}_t^l = \boldsymbol{H}_l(\boldsymbol{x}_t^l, h_{t-1}^l), \quad \boldsymbol{x}_t^l = [\boldsymbol{h}_t^{l-1}; \boldsymbol{x}_t^{l-1}]. \tag{2}$$

Note that $[.;.]$ denotes the concatenation operation. The outputs $\{\boldsymbol{h}_i^a | i = 1, 2, ..., l_a\}$ and $\{\boldsymbol{h}_j^b | j = 1, 2, ..., l_b\}$ of stack-GRU are the concatenation of outputs of all GRU layers and the inputs of Attention Stack-GRU unit for sentence a and b, which can preserve all the information, as well as the previous feature work in word embedding part. Figure 1 (B) shows the detailed structure.

However, this architecture only models the sentence and gathers all the information into vectors in a comprehensive way. How to compress these vectors into one sentence representation is still unclear. Since natural language has the redundancy mechanism (Luuk and Luuk 2011), different words have different contributions to the sentence semantic. Moreover, attention mechanism allows the model to focus on distinct aspects of the input and thus improve its ability to extract the most relevant parts for outputs (Cho, Courville,

and Bengio 2015). Therefore, it is natural to utilize attention mechanism to integrate the outputs of Stack-GRU:

$$\begin{aligned}
\boldsymbol{A}^a &= [\boldsymbol{h}_1^a, \boldsymbol{h}_2^a, ..., \boldsymbol{h}_{l_a}^a], \\
\boldsymbol{\alpha}^a &= \boldsymbol{\omega}^{\mathrm{T}} \tanh(\boldsymbol{W}\boldsymbol{A}^a + b), \\
\boldsymbol{h}^a &= \sum_{i=1}^{l_a} \frac{\exp(\alpha_i^a)}{\sum_{k=1}^{l_a} \exp(\alpha_k^a)} \boldsymbol{h}_i^a, \quad i = 1, 2, ..., l_a,
\end{aligned} \tag{3}$$

where $\boldsymbol{h}^a$ is original representation of sentence a, which is actually a weight summation of the final outputs of stack-GRU. $\boldsymbol{\omega}$, $\boldsymbol{W}$, and b are trainable parameters. The same operation will be done on sentence b to get the original representation $\boldsymbol{h}^b$. By utilizing this operation, *DRr-Net* can gather all the important information, which is critical for the sentence semantic, to generate the original sentence representation.

Dynamic Re-read Mechanism. Psychological researches have shown that people usually cannot pay close attention to too many things at the same time (Wang et al. 1999). In fact, humans could focus on less than 7 different object at the same time. When reading a piece of text intensively, people focus on only 1.5 words each time (Wang et al. 1999). Moreover, with an in-depth understanding of the sentence, the important words that should be concerned are dynamically changing, even the words that did not get attention before. In order to pay close attention to the most important word at each step with the consideration of learned information, we develop the Dynamic Re-read (DRr) Mechanism, as shown in the Figure 1 (C), the DRr unit selects the most important word at each step with the consideration of original representations, and the selections in previous steps.

To be specific, the inputs of Dynamic Re-read unit are the final outputs $\{h_i^a | i = 1, 2, ..., l_a\}$ and $\{h_i^b | i = 1, 2, ..., l_b\}$ of ASG unit. In each step, we adopt attention mechanism to choose the word for current input from the whole input sequence. Then, we utilize GRU to encode the chosen words and dynamic context around it:

$$\begin{aligned}
\bar{\boldsymbol{a}}_t &= \mathrm{F}([\boldsymbol{h}_1^a, \boldsymbol{h}_2^a, ..., \boldsymbol{h}_{l_a}^a], \bar{\boldsymbol{h}}_{t-1}^a, \boldsymbol{h}^b), \\
\bar{\boldsymbol{h}}_t^a &= \mathrm{GRU}(\bar{\boldsymbol{a}}_t, \bar{\boldsymbol{h}}_{t-1}^a), \quad t = 1, 2, ..., T, \\
\boldsymbol{v}^a &= \bar{\boldsymbol{h}}_T^a,
\end{aligned} \tag{4}$$

where $\boldsymbol{h}^b$ is the original representation of sentence b. T is the dynamic re-read length. $\boldsymbol{v}^a$ denotes the dynamic representation of sentence a. In order to better understand sentence semantic, we also employ the original representation $\boldsymbol{h}^a$ as the initial state of GRU for sentence a. The function F is the choosing function at each step, and we utilize attention mechanism to achieve this function, which can be formulated as follows:

$$\begin{aligned}
\bar{\boldsymbol{A}}^a &= [\boldsymbol{h}_1^a, \boldsymbol{h}_2^a, ..., \boldsymbol{h}_{l_a}^a], \\
\bar{\boldsymbol{m}}^a &= \omega_d^T \tanh(\boldsymbol{W}_d \bar{\boldsymbol{A}}^a + (\boldsymbol{U}_d \bar{\boldsymbol{h}}_{t-1}^a + \boldsymbol{M}_d \boldsymbol{h}^b) \otimes \boldsymbol{e}_{l_a}), \\
\bar{\boldsymbol{\alpha}}^a &= \sum_{i=1}^{l_a} \frac{\exp(\bar{m}_i^a)}{\sum_{k=1}^{l_a} \exp(\bar{m}_k^a)}, \\
\bar{\boldsymbol{a}}_t &= \boldsymbol{h}_j^a, \quad (j = \mathrm{Index}(\max(\bar{\boldsymbol{\alpha}}^a))),
\end{aligned} \tag{5}$$

where $\mathrm{Index}(\max(\bar{\boldsymbol{\alpha}}^a))$ denotes getting the corresponding index of the maximum value in the attention vector $\bar{\boldsymbol{\alpha}}^a$. ω_d, $\boldsymbol{W}_d$, $\boldsymbol{U}_d$ and $\boldsymbol{M}_d$ are trainable parameters. $\boldsymbol{e}_{l_a} \in \mathbb{R}^{l_a}$ is a row vector of 1. The outer product $(\boldsymbol{U}_d \bar{\boldsymbol{h}}_{t-1}^a + \boldsymbol{M}_d \boldsymbol{h}^b) \otimes \boldsymbol{e}_{l_a}$ means repeating $(\boldsymbol{U}_d \bar{\boldsymbol{h}}_{t-1}^a + \boldsymbol{M}_d \boldsymbol{h}^b)$ l_a times.

To be specific, we treat the whole outputs of sentence a in ASG unit $[\boldsymbol{h}_1^a, \boldsymbol{h}_2^a, ..., \boldsymbol{h}_{l_a}^a]$, original representation $\boldsymbol{h}^b$ of sentence b , and previous hidden states $\bar{\boldsymbol{h}}_{t-1}^a$ as the inputs of attention unit. As mentioned before, attention mechanism can help the model focus on the most relevant parts. Since our goal is to model the relation between two sentences, we choose the original representation $\boldsymbol{h}^b$ of sentence b as one of the inputs of attention mechanism for sentence a. Along this line, the most important word at the t^{th} time step will be selected with the consideration of the previous information and the information of sentence b. However, $\mathrm{Index}(\max(.))$ operation has no derivative, which means its gradient could not be calculated. Fortunately, our goal is to select the most important word, which requires one word at one step. Inspired by $\mathrm{softmax}$ function, we modify Eq.(5) as follows:

$$\begin{aligned}
\bar{\boldsymbol{A}}^a &= [\boldsymbol{h}_1^a, \boldsymbol{h}_2^a, ..., \boldsymbol{h}_{l_a}^a], \\
\bar{\boldsymbol{m}}^a &= \omega_d^T \tanh(\boldsymbol{W}_d \bar{\boldsymbol{A}}^a + (\boldsymbol{U}_d \bar{\boldsymbol{h}}_{t-1}^a + \boldsymbol{M}_d \boldsymbol{h}^b) \otimes \boldsymbol{e}_{l_a}), \\
\bar{\boldsymbol{a}}_t &= \sum_{i=1}^{l_a} \frac{\exp(\beta \bar{m}_i^a)}{\sum_{k=1}^{l_a} \exp(\beta \bar{m}_k^a)} \boldsymbol{h}_i^a,
\end{aligned} \tag{6}$$

where β is an arbitrarily big value. With this operation, the weight of the most important word will be very close to 1, and other weights will be very close to 0.

Label Prediction. This component consists of three operations: matching, fusion and classification. In order to determine the overall relation between two sentences, we leverage heuristic matching (Chen et al. 2017c) between original representations $\boldsymbol{h}^a$, $\boldsymbol{h}^b$ and dynamic representations $\boldsymbol{v}^a$, $\boldsymbol{v}^b$. Specifically, we use the element-wise product, their difference and concatenation. Then, we send them to multi-layer perceptron (MLP) to calculate the relation probability between the sentences pair. The MLP has two hidden layers with ReLu activation and a $\mathrm{softmax}$ output layer.

$$\begin{aligned}
\boldsymbol{h} &= (\boldsymbol{h}^a,\ \boldsymbol{h}^b,\ \boldsymbol{h}^b \odot \boldsymbol{h}^a,\ \boldsymbol{h}^b - \boldsymbol{h}^a), \\
\boldsymbol{v} &= (\boldsymbol{v}^a,\ \boldsymbol{v}^b,\ \boldsymbol{v}^b \odot \boldsymbol{v}^a,\ \boldsymbol{v}^b - \boldsymbol{v}^a), \\
\boldsymbol{p}^h &= \mathrm{MLP}_1(\boldsymbol{h}), \\
\boldsymbol{p}^v &= \mathrm{MLP}_1(\boldsymbol{v}),
\end{aligned} \tag{7}$$

where $\boldsymbol{p}^h$ and $\boldsymbol{p}^v$ denote the probability distribution of different classes with original sentence representations and dynamic sentence representations separately.

In matching operation, concatenation can retain all the information (Zhang et al. 2017). The element-wise product is a certain measure of "similarity" of two sentences (Mou et al. 2016). Their difference can capture the degree of distributional inclusion in each dimension (Weeds et al. 2014).

After getting the different probability distribution among the relations with different sentence semantic representations, we intend to integrate these information to achieve

Table 1: Performance (accuracy) of models on different SNLI test sets and SICK test set.

Model	#Paras	Full test	Hard test	Lexical test	SICK test
(1) CENN (Zhang et al. 2017)	≈800k	82.1%	60.4%	51.9%	81.8%
(2) BiLSTM with Inner-Attention (Liu et al. 2016)	2.8m	84.5%	62.7%	58.6%	85.2%
(3) Gated-Att BiLSTM (Chen et al. 2017b)	12m	85.5%	65.5%	65.6%	85.7%
(4) CAFE (Tay, Tuan, and Hui 2017)	3.7m	85.9%	66.1%	65.5%	86.1%
(5) Gumbel TreeLSTM (Choi, Yoo, and Lee 2018)	2.9m	86.0%	66.7%	67.3%	85.8%
(6) Distance-based SAN (Im and Cho 2017)	4.7m	86.3%	67.4%	68.5%	86.7%
(7) DRCN (Kim et al. 2018)	5.6m	86.5%	68.3%	69.4%	87.4%
(8) *DRr-Net*	3.5m	**87.7**%	**71.4**%	**76.5**%	**88.3**%

Table 2: Performance (accuracy) on Quora Question Pair.

Model	Accuracy
(1) CENN (Zhang et al. 2017)	80.72%
(2) MP-LSTM (Wang, Hamza, and Florian 2017)	83.21%
(3) L.D.C (Wang, Mi, and Ittycheriah 2016)	85.55%
(4) BiMPM (Wang, Hamza, and Florian 2017)	88.17%
(5) pt-DecAttachar.c (Tomar et al. 2017)	88.40%
(6) DIIN (Gong, Luo, and Zhang 2017)	89.06%
(7) *DRr-Net*	**89.75**%

Table 3: Ablation Performance (accuracy) of *DRr-Net*.

Model	SNLI test	SICK test
(1) *DRr-Net* (w/o initial operation)	85.3%	85.7%
(2) *DRr-Net* (w/o guide operation)	85.1%	86.1%
(3) *DRr-Net* (w/o matching operation)	83.2%	82.6%
(4) *DRr-Net* (w/o origin result)	81.2%	80.5%
(5) *DRr-Net* (w/o re-read result)	85.6%	86.4 %
(6) *DRr-Net*	**87.7**%	**88.3**%

more robust performance. Thus, we utilize a fusion gate and a multi-layer perceptron (MLP) to integrate them and make the final classification, which can be formulated as follows:

$$\begin{aligned} \alpha_h &= \sigma(\boldsymbol{w}_h^{\mathrm{T}}\boldsymbol{p}^h + b_h), \\ \alpha_v &= \sigma(\boldsymbol{w}_v^{\mathrm{T}}\boldsymbol{p}^v + b_v), \\ P(y|(\boldsymbol{s}^a, \boldsymbol{s}^b)) &= \mathrm{MLP}_2(\alpha_h\boldsymbol{p}^h + \alpha_v\boldsymbol{p}^v). \end{aligned} \tag{8}$$

3.3 Model Learning

In this subsection, we will introduce the details about the model learning, which can be classified into two parts: 1) Loss Function; 2) Model Initialization.

Loss Function: We employ the *cross-entropy* as the loss function since the goal is to make the correct classification. The following is the loss function for the output of last layer:

$$L^c = -\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{y}_i \log P(y_i|(\boldsymbol{s}_i^a, \boldsymbol{s}_i^b)), \tag{9}$$

where $\boldsymbol{y}_i$ is the one-hot representation for the true class of the i^{th} instance, and N represents the number of training instances. In order to make $\boldsymbol{p}^h, \boldsymbol{p}^v$ also calculate the correct probability distribution, we apply *cross-entropy* function to both of them. Considering the model complexity, we also add l2-norm of all trainable parameters of *DRr-Net* to the final loss function, which is formulated as follows:

$$L = L^h + L^v + L^c + \epsilon \left\|\boldsymbol{\theta}\right\|_2, \tag{10}$$

where $\boldsymbol{\theta}$ denotes all training parameters in the model.

Model Initialization: In order to get the best performance, we have tuned the hyper-parameters on the validation set. Their values are illustrated as follows:

We obtained the word embedding from a pre-trained word vectors (840B GloVe) (Pennington, Socher, and Manning 2014), which the dimension is set as 300. Character-level word embedding is set as 100. The number of stack layers in ASG unit is set as 3 and the re-read length in DRr unit is set as 6. The hidden state size of GRUs in these two units is set as 256. To initialize the model, we randomly set the all weights, such as $\boldsymbol{W}$, following the uniform distribution in the range between $-\sqrt{6/(nin+nout)}$ and $\sqrt{6/(nin+nout)}$ as suggested by (Orr and Müller 2003). All biases such as $\boldsymbol{b}$ are set as zeros. We use Adam optimizer with learning rate 10^{-4}. During implementation, we utilize *Photinia*[1] to build our entire model.

4 Experiment

In this section, we first evaluate the model performance on three benchmark datasets for two challenging sentence semantic matching tasks: 1) SNLI and SICK for natural language inference; 2) Quora Question Pair for paraphrase identification. Then, we give a detailed analysis of the model and experiment results.

4.1 Data Description

We evaluate our model on three well-studied datasets: the Stanford Natural Language Inference (SNLI), the Sentence Involving Compositional Knowledge (SICK), and Quora duplicate questions (Quora).

SNLI. The SNLI (Bowman et al. 2015) contains $570,152$ human annotated sentence pairs. Each sentence pair is labeled with one of the following relations: *Entailment*, *Contradiction*, or *Neutral*.

SICK. The SICK (Marelli et al. 2014) contains $10,000$ sentence pairs. The labels are the same as SNLI dataset.

Quora. The Quora Question Pair (Iyer, Dandekar, and Csernai 2017) dataset consists of over $400,000$ potential question duplicate pairs. Each pair has a binary value that indicates whether the line truly contains a duplicate pair.

[1] https://github.com/XoriieInpottn/photinia

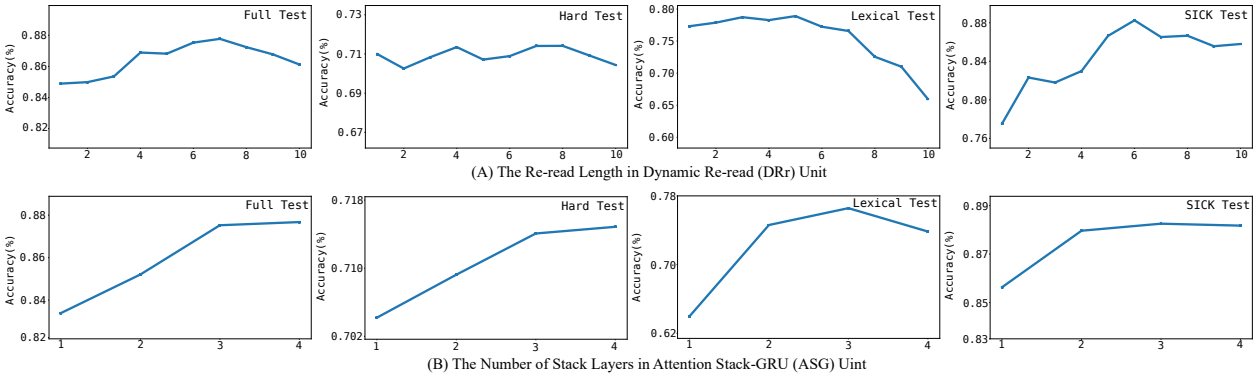


Figure 2: Performance (accuracy) of *DRr-Net* with different reading lengths (1-10) or the number stack layers (1-4).

4.2 Experiment Results

We evaluate models on the NLI task over SNLI and SICK datasets. In order to reduce the impact of annotate artifacts and better evaluate their ability of sentence understanding, we also select the challenging hard subset from (Gururangan et al. 2018), in which the premise-oblivious model cannot classify accurately, and lexical subset from (Glockner, Shwartz, and Goldberg 2018), which requires lexical and world knowledge, as our SNLI test sets.

Table 1 shows the results of our model compared with other published sentence encoding-based models. We utilize the accuracy on test sets to evaluate their performances.

From Table 1, we can figure out that *DRr-Net* achieves the best performance on all test sets. As described in Section 3, *DRr-Net* utilizes ASG unit to model sentence semantic comprehensively by repeatedly reading the sentence and preserving all the information from bottom-most word embedding input to up-most recurrent output. Therefore, the sentence semantic can be fully explored. Moreover, DRr unit is employed to pay close attention to one important word at each step and repeatedly reads the important words for better sentence semantic understanding. Thus, *DRr-Net* achieves the best performance on all the test sets, even the challenging hard test and lexical test sets.

Among these sentence encoding-based baselines, DRCN and Distance-based SAN are the current state-of-the-art models. Distance-based SAN utilizes the masked multi-head attention with distance to model the sentence semantic, which can effectively explore the sentence semantic from multiple aspects. DRCN adopts densely-connected co-attentive network to generate sentence representations. It can preserve the original and co-attentive feature information among the entire architecture. However, each attention operation in these two methods considers too much information at one time, which leads the models unable to focus on the most relevant part precisely as *DRr-Net* did.

Gururangan et al. (2018) suspects that annotation artifacts inflate model performance. Thus, they propose a challenging hard subset of SNLI to better evaluate the models' ability on sentence semantic. Since the examples that premise-oblivious model classified accurately are removed, this test set can focus the model on sentence semantic rather than the annotate artifacts and better evaluate their performances. From the results in Table 1, we can conclude that *DRr-Net* outperforms all baselines by a large margin, e.g. Distance-based DRCN (+3.1%), Distance-based SAN (+4.0%) and CAFE (+5.3%). We can find the same phenomenon on the challenging lexical test set. All of these indicate that our proposed model had better adaptability.

Table 2 shows performances of all models on the Quora dataset. BiMPM using the multi-perspective matching technique between two sentences reports baseline performance of a L.D.C. network and basic multi-perspective models. We obtain accuracy of 89.75% on this dataset, surpassing the previous state-of-the-art model of DIIN.

4.3 Ablation Performance

We conduct an ablation study on *DRr-Net* to examine the effectiveness of each component. The results are illustrated in Table 3. As mentioned before, we utilize the original representation $\boldsymbol{h}^a$ as the initial value of GRU in its DRr unit, and the original representation $\boldsymbol{h}^b$ as one of the inputs of attention unit in its DRr unit. These operations make *DRr-Net* have a comprehensive understanding of sentence a and search the most relevant part with the consideration of sentence b and learned information at each step. As shown in Table 3 (1)-(2), the performance of *DRr-Net* significantly decreased when removing them separately, which means the initial operation and guide operation are critical for deciding which part should be concerned more at each step.

Recalling the model architecture, the original result and dynamic re-read result are fused in the Label Prediction part. We are curious whether only the original result or dynamic re-read result is enough for classification. Thus, we remove them separately to verify it. The results in Table 3 (4)-(5) illustrate that both parts are extremely important for classification. ASG unit can model the sentence semantic in a comprehensive way. However, it cannot focus on the every important part precisely. DRr unit is capable of focusing on the most important part at each step, but lacking a comprehensive understanding of original sentences. In other words, both of them are indispensable for *DRr-Net* to achieve better performance in sentence semantic matching.

Table 4: Some examples of re-read sequence and the classification.

Sentence	Re-read sequence	Gold	Predicted
s^a: a couple walk hand in hand down a street.	walk walk couple couple street street	Contradiction	Contradiction
s^b: a couple is sitting on a bench.	couple sitting sitting bench bench bench		
s^a: a person in a red shirt and black pants hunched over.	red shirt red shirt red shirt	Entailment	Entailment
s^b: a person wears a red shirt.	wears red shirt red shirt red		
s^a: a person with a purple shirt is painting an image of a woman on a white wall.	person painting painting woman woman woman	Neutral	Neutral
s^b: a woman paints a portrait of her best friend.	woman paints paints best best best		
s^a: the man in the black t-shirt is trying to throw something.	black t-shirt is trying throw throw	Entailment	Neutral
s^b: the man is in a black shirt.	black black shirt shirt shirt shirt		
s^a: a person with a purple shirt is painting an image of a woman on a white wall.	person painting painting woman woman woman	Neutral	Entailment
s^b: a woman paints a portrait of a person.	paints portrait portrait portrait portrait person		

4.4 Sensitivity of Parameters

As mentioned in Section 3, there are two parameters in *DRr-Net* should be determined: the number of stack layers in ASG unit and the re-read length in DRr unit. To be specific, we evaluate the performances of *DRr-Net* on SNLI and SICK datasets with different number of stack layers and re-read length separately. The results are shown in Figure 2

Figure 2 (A) shows the results on different test sets with different re-read length. The performance of *DRr-Net* first becomes better with the increasing of re-read length. When the re-read length is between 5 to 7, *DRr-Net* achieves the best performance. This phenomenon is consistent with the psychological findings that human attention focuses on nearly 7 words (Tononi 2008). When the length is bigger than 7, the accuracy of *DRr-Net* decreases to varying degrees, in which the accuracy on lexical test set dropped most. Recalling the data collection, the sentence pair in lexical test set has high overlap. Only one word is different in each sentence pair. Therefore, when the re-read length becomes too long, the model may lose the focus on the different words, which leads a wrong classification.

Figure 2 (B) illustrates the results with different number of stack layers in ASG Unit. We can conclude that the performance becomes better with the increasing of the number of stack layers from the result. However, as illustrated in Figure 1 (B), the inputs of the t^{th} layer are the concatenation of the inputs and outputs of the $(t-1)^{th}$ layer, which means the scale of parameters will grow rapidly with the increasing of the number of stack layer. However, the increasing rate of accuracy will slow down with the increasing of the number of stack layer in Figure 2 (B). Moreover, a large number of parameters may cause the model hard to optimized, and even worse the gradient might be exploded or vanished. Thus, we select 3 layers in the ASG unit.

4.5 Case Study and Error Analysis

We also show some examples from SNLI dataset to demonstrate the ability of *DRr-Net* of dynamic re-reading the whole sequence. Table 4 shows the important word that the model chose at each step. For the first example, *DRr-Net* pays attention to words "*walk, couple, street*" in premise and "*couple, sitting, bench*" in hypothesis. Then, *DRr-Net* repeatedly processes these important words for final decision. From these words, we can conclude that the relation of this sentence pair was contradiction easily. Moreover, when checking the entailment relation in the second example, *DRr-Net* processes the same important words repeatedly, i.e., reading "red shirt" multiple times. In a word, *DRr-Net* does really choose the important region and re-read these important parts multiple time for the final decision.

In order to better verify the ability of *DRr-Net*, we make error analysis on the misclassification examples. In the forth example, when sentence a contains more information than sentence b, *DRr-Net* may consider the unseen information in sentence b more and make a wrong classification. Moreover, when the sentence pair has very complex semantic relation, e.g., the last example in Table 4, the model may be confused about their semantic and suffer from one of the important words, which leads to a wrong classification result.

Since sentence semantic suffers from the issues such as polysemy, ambiguity, as well as fuzziness. The model may need more information to distinguish these relations and make the correct decision.

5 Conclusion and Future Work

In this paper, we proposed a *Dynamic Re-read Network (DRr-Net)* approach for sentence semantic matching, a novel architecture that was able to pay close attention to a small region of sentences at each time and re-read the important information for better sentence semantic matching. To be specific, we employed Attention Stack-GRU (ASG) unit to model sentence semantic comprehensively and preserve all the learned information. Then, we utilized Dynamic Re-read (DRr) mechanism to pay close attention to one important word at each step with the consideration of learned information and re-read the important words for better sentence semantic understanding. Extensive experiments on three benchmark sentence matching datasets demonstrated the superiority of our proposed model. Moreover, the length setting of *DRr-Net* was consistent with the findings of psychological researches. In the future, we will focus on providing more information for attention mechanism to select important part more precisely and reduce the situation of repeated reading of one word.

6 Acknowledgements

This research was partially supported by grants from the National Key Research and Development Program of China (No.2016YFB1000904, 2017YFB0803301), and the National Natural Science Foundation of China (Grants No. 61727809, U1605251, 61602147).

References

Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*, 632–642.

Chen, D.; Fisch, A.; Weston, J.; and Bordes, A. 2017a. Reading wikipedia to answer open-domain questions. In *ACL*, 1870–1879.

Chen, Q.; Zhu, X.-D.; Ling, Z.-H.; Wei, S.; Jiang, H.; and Inkpen, D. 2017b. Recurrent neural network-based sentence encoder with gated attention for natural language inference. In *RepEval@EMNLP*, 36–40.

Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; Jiang, H.; and Inkpen, D. 2017c. Enhanced lstm for natural language inference. In *ACL*.

Cheng, J.; Dong, L.; and Lapata, M. 2016. Long short-term memory-networks for machine reading. In *EMNLP*.

Cho, K.; Courville, A. C.; and Bengio, Y. 2015. Describing multimedia content using attention-based encoder-decoder networks. *IEEE Trans. Multimedia* 17:1875–1886.

Choi, J.; Yoo, K. M.; and Lee, S.-g. 2018. Learning to compose task-specific tree structures. In *AAAI*.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR* abs/1412.3555.

Clark, P.; Etzioni, O.; Khot, T.; Sabharwal, A.; Tafjord, O.; Turney, P. D.; and Khashabi, D. 2016. Combining retrieval, statistics, and inference to answer elementary science questions. In *AAAI*.

Dolan, W. B., and Brockett, C. 2005. Automatically constructing a corpus of sentential paraphrases. In *IWP*.

Glockner, M.; Shwartz, V.; and Goldberg, Y. 2018. Breaking nli systems with sentences that require simple lexical inferences. In *ACL*, 650–655.

Gong, Y.; Luo, H.; and Zhang, J. 2017. Natural language inference over interaction space. *CoRR* abs/1709.04348.

Gururangan, S.; Swayamdipta, S.; Levy, O.; Schwartz, R.; Bowman, S. R.; and Smith, N. A. 2018. Annotation artifacts in natural language inference data. In *NAACL-HLT*, 107–112.

Im, J., and Cho, S. 2017. Distance-based self-attention network for natural language inference. *CoRR* abs/1712.02047.

Iyer, S.; Dandekar, N.; and Csernai, K. 2017. First quora dataset release: Question pairs.

Kim, S.; Hong, J.-H.; Kang, I.; and Kwak, N. 2018. Semantic sentence matching with densely-connected recurrent and co-attentive information. *CoRR* abs/1805.11360.

Koch, C., and Tsuchiya, N. 2007. Attention and consciousness: two distinct brain processes. *Trends in cognitive sciences* 11(1):16–22.

Kun, Z.; Guangyi, L.; Le, W.; Enhong, C.; Qi, L.; and Han, W. 2018. Image-enhanced multi-level sentence representation net for natural language inference. In *ICDM*.

Liu, Y.; Sun, C.; Lin, L.; and Wang, X. 2016. Learning natural language inference using bidirectional lstm model and inner-attention. *CoRR* abs/1605.09090.

Liu, Q.; Huang, Z.; Huang, Z.; Liu, C.; Chen, E.; Su, Y.; and Hu, G. 2018. Finding similar exercises in online education systems. In *SIGKDD*, 1821–1830. ACM.

Luuk, E., and Luuk, H. 2011. The redundancy of recursion and infinity for natural language. *Cognitive Processing* 12(1):1–11.

Marelli, M.; Bentivogli, L.; Baroni, M.; Bernardi, R.; Menini, S.; and Zamparelli, R. 2014. Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *SemEval*, 1–8.

Mou, L.; Men, R.; Li, G.; Xu, Y.; Zhang, L.; Yan, R.; and Jin, Z. 2016. Natural language inference by tree-based convolution and heuristic matching. In *ACL*, volume 2, 130–136.

Nie, Y., and Bansal, M. 2017. Shortcut-stacked sentence encoders for multi-domain inference. In *RepEval@EMNLP*.

Orr, G. B., and Müller, K.-R. 2003. *Neural networks: tricks of the trade*. Springer.

O'Shea, M. 1908. The psychology and pedagogy of reading. *The Journal of Philosophy, Psychology and Scientific Methods* 5(18):500–502.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *EMNLP*, 1532–1543.

Serban, I. V.; Sordoni, A.; Bengio, Y.; Courville, A. C.; and Pineau, J. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, volume 16.

Shen, T.; Zhou, T.; Long, G.; Jiang, J.; Pan, S.; and Zhang, C. 2017. Disan: Directional self-attention network for rnn/cnn-free language understanding. *CoRR* abs/1709.04696.

Srivastava, R. K.; Greff, K.; and Schmidhuber, J. 2015. Highway networks. *CoRR* abs/1505.00387.

Tay, Y.; Tuan, L. A.; and Hui, S. C. 2017. A compare-propagate architecture with alignment factorization for natural language inference. *CoRR* abs/1801.00102.

Tomar, G. S.; Duque, T.; Täckström, O.; Uszkoreit, J.; and Das, D. 2017. Neural paraphrase identification of questions with noisy pretraining. In *SWCN@EMNLP*.

Tononi, G. 2008. Consciousness as integrated information: a provisional manifesto. *The Biological Bulletin* 215(3):216–242.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NIPS*, 5998–6008.

Wang, J.; Chen, H.-C.; Radach, R.; and Inhoff, A. 1999. *Reading Chinese script: A cognitive analysis*. Psychology Press.

Wang, P.; Wu, Q.; Shen, C.; and van den Hengel, A. 2017. The vqa-machine: Learning how to use existing vision algorithms to answer new questions. In *CVPR*, volume 4.

Wang, Z.; Hamza, W.; and Florian, R. 2017. Bilateral multi-perspective matching for natural language sentences. *CoRR* abs/1702.03814.

Wang, Z.; Mi, H.; and Ittycheriah, A. 2016. Sentence similarity learning by lexical decomposition and composition. In *COLING*.

Weeds, J.; Clarke, D.; Reffin, J.; Weir, D.; and Keller, B. 2014. Learning to distinguish hypernyms and co-hyponyms. In *COLING*.

Yarbus, A. L. 1967. Eye movements during perception of complex objects. In *Eye movements and vision*. Springer. 171–211.

Zhang, K.; Chen, E.; Liu, Q.; Liu, C.; and Lv, G. 2017. A context-enriched neural network method for recognizing lexical entailment. In *AAAI*, 3127–3134.