

# DR-BiLSTM: Dependent Reading Bidirectional LSTM for Natural Language Inference

Reza Ghaeini<sup>1\*</sup>, Sadid A. Hasan<sup>2</sup>, Vivek Datla<sup>2</sup>, Joey Liu<sup>2</sup>, Kathy Lee<sup>2</sup>, Ashequl Qadir<sup>2</sup>, Yuan Ling<sup>2</sup>, Aaditya Prakash<sup>2</sup>, Xiaoli Z. Fern<sup>1</sup> and Oladimeji Farri<sup>2</sup>

<sup>1</sup>Oregon State University, Corvallis, OR, USA

<sup>2</sup>Artificial Intelligence Laboratory, Philips Research North America, Cambridge, MA, USA

{ghaeinim,xfern}@eecs.oregonstate.edu

{sadid.hasan,vivek.datla,joey.liu,kathy.lee\_1,ashequl.qadir}@philips.com

{yuan.ling,aaditya.prakash,dimeji.farri}@philips.com

## Abstract

We present a novel deep learning architecture to address the natural language inference (NLI) task. Existing approaches mostly rely on simple reading mechanisms for independent encoding of the premise and hypothesis. Instead, we propose a novel *dependent reading* bidirectional LSTM network (DR-BiLSTM) to efficiently model the relationship between a premise and a hypothesis during encoding and inference. We also introduce a sophisticated ensemble strategy to combine our proposed models, which noticeably improves final predictions. Finally, we demonstrate how the results can be improved further with an additional pre-processing step. Our evaluation shows that DR-BiLSTM obtains the best single model and ensemble model results achieving the new state-of-the-art scores on the Stanford NLI dataset.

## 1 Introduction

Natural Language Inference (NLI; a.k.a. Recognizing Textual Entailment, or RTE) is an important and challenging task for natural language understanding (MacCartney and Manning, 2008). The goal of NLI is to identify the logical relationship (*entailment*, *neutral*, or *contradiction*) between a premise and a corresponding hypothesis. Table 1 shows few example relationships from the Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015).

Recently, NLI has received a lot of attention from the researchers, especially due to the availability of large annotated datasets like SNLI (Bow-

<b>P<sup>a</sup></b>	A senior is waiting at the window of a restaurant that serves sandwiches.	Relationship
<b>H<sup>b</sup></b>	A person waits to be served his food.	Entailment
	A man is looking to order a grilled cheese sandwich.	Neutral
	A man is waiting in line for the bus.	Contradiction
<sup>a</sup> P, Premise. <sup>b</sup> H, Hypothesis.		

Table 1: Examples from the SNLI dataset.

man et al., 2015). Various deep learning models have been proposed that achieve successful results for this task (Gong et al., 2017; Wang et al., 2017; Chen et al., 2017; Yu and Munkhdalai, 2017a; Parikh et al., 2016; Zhao et al., 2016; Sha et al., 2016). Most of these existing NLI models use attention mechanism to jointly interpret and align the premise and hypothesis. Such models use simple reading mechanisms to encode the premise and hypothesis independently. However, such a complex task require explicit modeling of dependency relationships between the premise and the hypothesis during the encoding and inference processes to prevent the network from the loss of relevant, contextual information. In this paper, we refer to such strategies as *dependent reading*.

There are some alternative reading mechanisms available in the literature (Sha et al., 2016; Rocktäschel et al., 2015) that consider dependency aspects of the premise-hypothesis relationships. However, these mechanisms have two major limitations:

- So far, they have only explored dependency aspects during the encoding stage, while ignoring its benefit during inference.
- Such models only consider encoding a hypothesis depending on the premise, disre-

\*This work was conducted as part of an internship program at Philips Research.

garding the dependency aspects in the opposite direction.

We propose a dependent reading bidirectional LSTM (DR-BiLSTM) model to address these limitations. Given a premise  $u$  and a hypothesis  $v$ , our model first encodes them considering dependency on each other ( $u|v$  and  $v|u$ ). Next, the model employs a soft attention mechanism to extract relevant information from these encodings. The augmented sentence representations are then passed to the inference stage, which uses a similar dependent reading strategy in both directions, i.e.  $u \rightarrow v$  and  $v \rightarrow u$ . Finally, a decision is made through a multi-layer perceptron (MLP) based on the aggregated information.

Our experiments on the SNLI dataset show that DR-BiLSTM achieves the best single model and ensemble model performance obtaining improvements of a considerable margin of 0.4% and 0.3% over the previous state-of-the-art single and ensemble models, respectively.

Furthermore, we demonstrate the importance of a simple preprocessing step performed on the SNLI dataset. Evaluation results show that such preprocessing allows our single model to achieve the same accuracy as the state-of-the-art ensemble model and improves our ensemble model to outperform the state-of-the-art ensemble model by a remarkable margin of 0.7%. Finally, we perform an extensive analysis to clarify the strengths and weaknesses of our models.

## 2 Related Work

Early studies use small datasets while leveraging lexical and syntactic features for NLI (MacCartney and Manning, 2008). The recent availability of large-scale annotated datasets (Bowman et al., 2015; Williams et al., 2017) has enabled researchers to develop various deep learning-based architectures for NLI.

Parikh et al. (2016) propose an attention-based model (Bahdanau et al., 2014) that decomposes the NLI task into sub-problems to solve them in parallel. They further show the benefit of adding intra-sentence attention to input representations. Chen et al. (2017) explore sequential inference models based on chain LSTMs with attentional input encoding and demonstrate the effectiveness of syntactic information. We also use similar attention mechanisms. However, our model is distinct

from these models as they do not benefit from dependent reading strategies.

Rocktäschel et al. (2015) use a word-by-word neural attention mechanism while Sha et al. (2016) propose re-read LSTM units by considering the dependency of a hypothesis on the information of its premise ( $v|u$ ) to achieve promising results. However, these models suffer from weak inferencing methods by disregarding the dependency aspects from the opposite direction ( $u|v$ ). Intuitively, when a human judges a premise-hypothesis relationship, s/he might consider back-and-forth reading of both sentences before coming to a conclusion. Therefore, it is essential to encode the premise-hypothesis dependency relations from both directions to optimize the understanding of their relationship.

Wang et al. (2017) propose a bilateral multi-perspective matching (BiMPM) model, which resembles the concept of matching a premise and hypothesis from both directions. Their matching strategy is essentially similar to our attention mechanism that utilizes relevant information from the other sentence for each word sequence. They use similar methods as Chen et al. (2017) for encoding and inference, without any dependent reading mechanism.

Although NLI is well studied in the literature, the potential of dependent reading and interaction between a premise and hypothesis is not rigorously explored. In this paper, we address this gap by proposing a novel deep learning model (DR-BiLSTM). Experimental results demonstrate the effectiveness of our model.

## 3 Model

Our proposed model (DR-BiLSTM) is composed of the following major components: input encoding, attention, inference, and classification. Figure 1 demonstrates a high-level view of our proposed NLI framework.

Let  $u = [u_1, \dots, u_n]$  and  $v = [v_1, \dots, v_m]$  be the given premise with length  $n$  and hypothesis with length  $m$  respectively, where  $u_i, v_j \in \mathbb{R}^r$  is an word embedding of  $r$ -dimensional vector. The task is to predict a label  $y$  that indicates the logical relationship between premise  $u$  and hypothesis  $v$ .

### 3.1 Input Encoding

RNNs are the natural solution for variable length sequence modeling, consequently, we utilize a

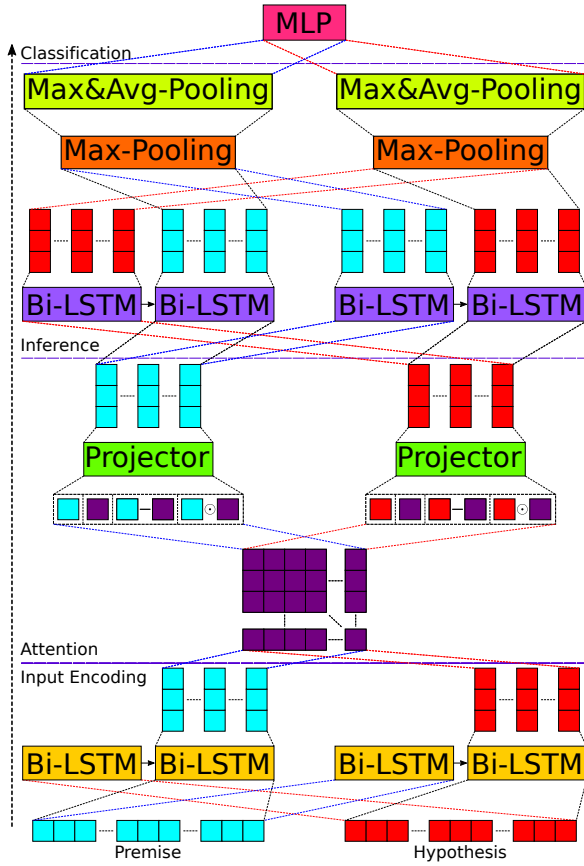


Figure 1: A high-level view of DR-BiLSTM. The data (premise  $u$  and hypothesis  $v$ , depicted with cyan and red tensors respectively) flows from bottom to top. Relevant tensors are shown with the same color and elements with the same colors share parameters.

bidirectional LSTM (BiLSTM) (Hochreiter and Schmidhuber, 1997) for encoding the given sentences. For ease of presentation, we only describe how we encode  $u$  depending on  $v$ . The same procedure is utilized for the reverse direction ( $v|u$ ).

To dependently encode  $u$ , we first process  $v$  using the BiLSTM. Then we read  $u$  through the BiLSTM that is initialized with previous reading final states (memory cell and hidden state). Here we represent a word (e.g.  $u_i$ ) and its context depending on the other sentence (e.g.  $v$ ). Equations 1 and 2 formally represent this component.

$$\begin{aligned} \bar{v}, s_v &= \text{BiLSTM}(v, 0) \\ \hat{u}, - &= \text{BiLSTM}(u, s_v) \end{aligned} \quad (1)$$

$$\begin{aligned} \bar{u}, s_u &= \text{BiLSTM}(u, 0) \\ \hat{v}, - &= \text{BiLSTM}(v, s_u) \end{aligned} \quad (2)$$

where  $\{\bar{u} \in \mathbb{R}^{n \times 2d}, \hat{u} \in \mathbb{R}^{n \times 2d}, s_u\}$  and  $\{\bar{v} \in \mathbb{R}^{m \times 2d}, \hat{v} \in \mathbb{R}^{m \times 2d}, s_v\}$  are the independent reading sequences, dependent reading sequences, and BiLSTM final state of independent reading of  $u$  and  $v$  respectively. Note that, “-” in these equations means that we do not care about the associated variable and its value. BiLSTM inputs are the word embedding sequences and initial state vectors.  $\hat{u}$  and  $\hat{v}$  are passed to the next layer as the output of the input encoding component.

The proposed encoding mechanism yields a richer representation for both premise and hypothesis by taking the history of each other into account. Using a max or average pooling over the independent and dependent readings does not further improve our model. This was expected since dependent reading produces more promising and relevant encodings.

### 3.2 Attention

We employ a soft alignment method to associate the relevant sub-components between the given premise and hypothesis. In deep learning models, such purpose is often achieved with a soft attention mechanism. Here we compute the unnormalized attention weights as the similarity of hidden states of the premise and hypothesis with Equation 3 (energy function).

$$e_{ij} = \hat{u}_i \hat{v}_j^T, \quad i \in [1, n], j \in [1, m] \quad (3)$$

where  $\hat{u}_i$  and  $\hat{v}_j$  are the dependent reading hidden representations of  $u$  and  $v$  respectively which are computed earlier in Equations 1 and 2. Next, for each word in either premise or hypothesis, the relevant semantics in the other sentence is extracted and composed according to  $e_{ij}$ . Equations 4 and 5 provide formal and specific details of this procedure.

$$\tilde{u}_i = \sum_{j=1}^m \frac{\exp(e_{ij})}{\sum_{k=1}^m \exp(e_{ik})} \hat{v}_j, \quad i \in [1, n] \quad (4)$$

$$\tilde{v}_j = \sum_{i=1}^n \frac{\exp(e_{ij})}{\sum_{k=1}^n \exp(e_{kj})} \hat{u}_i, \quad j \in [1, m] \quad (5)$$

where  $\tilde{u}_i$  represents the extracted relevant information of  $\hat{v}$  by attending to  $\hat{u}_i$  while  $\tilde{v}_j$  represents the extracted relevant information of  $\hat{u}$  by attending to  $\hat{v}_j$ .

To further enrich the collected attentional information, a trivial next step would be to pass the concatenation of the tuples  $(\hat{u}_i, \tilde{u}_i)$  or  $(\hat{v}_j, \tilde{v}_j)$  which provides a linear relationship between them. However, the model would suffer from the absence of *similarity* and *closeness* measures. Therefore, we calculate the difference and element-wise product for the tuples  $(\hat{u}_i, \tilde{u}_i)$  and  $(\hat{v}_j, \tilde{v}_j)$  that represent the similarity and closeness information respectively (Chen et al., 2017; Kumar et al., 2016).

The difference and element-wise product are then concatenated with the computed vectors,  $(\hat{u}_i, \tilde{u}_i)$  or  $(\hat{v}_j, \tilde{v}_j)$ , respectively. Finally, a feed-forward neural layer with ReLU activation function projects the concatenated vectors from  $8d$ -dimensional vector space into a  $d$ -dimensional vector space (Equations 6 and 7). This helps the model to capture deeper dependencies between the sentences besides lowering the complexity of vector representations.

$$\begin{aligned} a_i &= [\hat{u}_i, \tilde{u}_i, \hat{u}_i - \tilde{u}_i, \hat{u}_i \odot \tilde{u}_i] \\ p_i &= \text{ReLU}(W_p a_i + b_p) \end{aligned} \quad (6)$$

$$\begin{aligned} b_j &= [\hat{v}_j, \tilde{v}_j, \hat{v}_j - \tilde{v}_j, \hat{v}_j \odot \tilde{v}_j] \\ q_j &= \text{ReLU}(W_p b_j + b_p) \end{aligned} \quad (7)$$

Here  $\odot$  stands for element-wise product while  $W_p \in \mathbb{R}^{8d \times d}$  and  $b_p \in \mathbb{R}^d$  are the trainable weights and biases of the projector layer respectively.

### 3.3 Inference

During this phase, we use another BiLSTM to aggregate the two sequences of computed matching vectors,  $p$  and  $q$  from the attention stage (Section 3.2). This aggregation is performed in a sequential manner to avoid losing effect of latent variables that might rely on the sequence of matching vectors.

Instead of aggregating the sequences of matching vectors individually, we propose a similar dependent reading approach for the inference stage. We employ a BiLSTM reading process (Equations 8 and 9) similar to the input encoding step discussed in Section 3.1. But rather than passing just the dependent reading information to the next step, we feed both independent reading ( $\bar{p}$  and  $\bar{q}$ ) and dependent reading ( $\hat{p}$  and  $\hat{q}$ ) to a max pooling layer, which selects maximum values from

each sequence of independent and dependent readings ( $\bar{p}_i$  and  $\hat{p}_i$ ) as shown in Equations 10 and 11. The main intuition behind this architecture is to maximize the inferencing ability of the model by considering both independent and dependent readings.

$$\begin{aligned} \bar{q}, s_q &= \text{BiLSTM}(q, 0) \\ \hat{p}, - &= \text{BiLSTM}(p, s_q) \end{aligned} \quad (8)$$

$$\begin{aligned} \bar{p}, s_p &= \text{BiLSTM}(p, 0) \\ \hat{q}, - &= \text{BiLSTM}(q, s_p) \end{aligned} \quad (9)$$

$$\tilde{p} = \text{MaxPooling}(\bar{p}, \hat{p}) \quad (10)$$

$$\tilde{q} = \text{MaxPooling}(\bar{q}, \hat{q}) \quad (11)$$

Here  $\{\bar{p} \in \mathbb{R}^{n \times 2d}, \hat{p} \in \mathbb{R}^{n \times 2d}, s_p\}$  and  $\{\bar{q} \in \mathbb{R}^{m \times 2d}, \hat{q} \in \mathbb{R}^{m \times 2d}, s_q\}$  are the independent reading sequences, dependent reading sequences, and BiLSTM final state of independent reading of  $p$  and  $q$  respectively. BiLSTM inputs are the word embedding sequences and initial state vectors.

Finally, we convert  $\tilde{p} \in \mathbb{R}^{n \times 2d}$  and  $\tilde{q} \in \mathbb{R}^{m \times 2d}$  to fixed-length vectors with pooling,  $U \in \mathbb{R}^{4d}$  and  $V \in \mathbb{R}^{4d}$ . As shown in Equations 12 and 13, we employ both max and average pooling and describe the overall inference relationship with concatenation of their outputs.

$$U = [\text{MaxPooling}(\tilde{p}), \text{AvgPooling}(\tilde{p})] \quad (12)$$

$$V = [\text{MaxPooling}(\tilde{q}), \text{AvgPooling}(\tilde{q})] \quad (13)$$

### 3.4 Classification

Here, we feed the concatenation of  $U$  and  $V$  ( $[U, V]$ ) into a multilayer perceptron (MLP) classifier that includes a hidden layer with *tanh* activation and *softmax* output layer. The model is trained in an end-to-end manner.

$$\text{Output} = \text{MLP}([U, V]) \quad (14)$$

## 4 Experiments and Evaluation

### 4.1 Dataset

The Stanford Natural Language Inference (SNLI) dataset contains 570K human annotated sentence pairs. The premises are drawn from the



Flickr30k (Plummer et al., 2015) corpus, and then the hypotheses are manually composed for each relationship class (*entailment*, *neutral*, *contradiction*, and *-*). The “-” class indicates that there is no consensus decision among the annotators, consequently, we remove them during the training and evaluation following the literature. We use the same data split as provided in Bowman et al. (2015) to report comparable results with other models.

## 4.2 Experimental Setup

We use pre-trained 300-*D* Glove 840*B* vectors (Pennington et al., 2014) to initialize our word embedding vectors. All hidden states of BiLSTMs during input encoding and inference have 450 dimensions ( $r = 300$  and  $d = 450$ ). The weights are learned by minimizing the log-loss on the training data via the Adam optimizer (Kingma and Ba, 2014). The initial learning rate is 0.0004. To avoid overfitting, we use dropout (Srivastava et al., 2014) with the rate of 0.4 for regularization, which is applied to all feedforward connections. During training, the word embeddings are updated to learn effective representations for the NLI task. We use a fairly small batch size of 32 to provide more exploration power to the model. Our observation indicates that using larger batch sizes hurts the performance of our model.

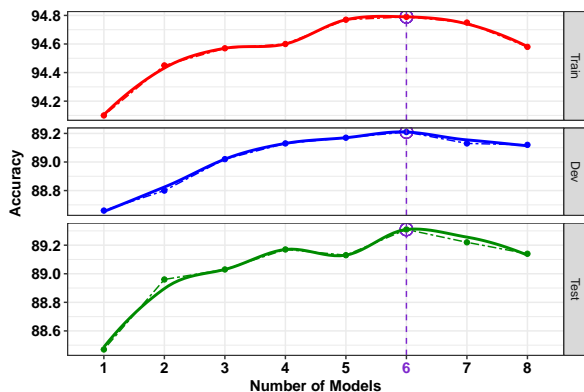


Figure 2: Performance of  $n$  ensemble models reported for training (red, top), development (blue, middle), and test (green, bottom) sets of SNLI. For  $n$  number of models, the best performance on the development set is used as the criteria to determine the final ensemble. The best performance on development set (89.22%) is observed using 6 models and is henceforth considered as our final DR-BiLSTM (Ensemble) model.

## 4.3 Ensemble Strategy

Ensemble methods use multiple models to obtain better predictive performance. Previous works typically utilize trivial ensemble strategies by either using majority votes or averaging the probability distributions over the same model with different initialization seeds (Wang et al., 2017; Gong et al., 2017).

By contrast, we use weighted averaging of the probability distributions where the weight of each model is learned through its performance on the SNLI development set. Furthermore, the differences between our models in the ensemble originate from: 1) variations in the number of dependent readings (i.e. 1 and 3 rounds of dependent reading), 2) projection layer activation (*tanh* and *ReLU* in Equations 6 and 7), and 3) different initialization seeds.

The main intuition behind this design is that the effectiveness of a model may depend on the complexity of a premise-hypothesis instance. For a simple instance, a simple model could perform better than a complex one, while a complex instance may need further consideration toward disambiguation. Consequently, using models with different rounds of dependent readings in the encoding stage should be beneficial.

Figure 2 demonstrates the observed performance of our ensemble method with different number of models. The performance of the models are reported based on the best obtained accuracy on the development set. We also study the effectiveness of other ensemble strategies e.g. majority voting, and averaging the probability distributions. But, our ensemble strategy performs the best among them (see Section A in the Appendix for additional details).

## 4.4 Preprocessing

We perform a trivial preprocessing step on SNLI to recover some out-of-vocabulary words found in the development set and test set. Note that our vocabulary contains all words that are seen in the training set, so there is no out-of-vocabulary word in it. The SNLI dataset is not immune to human errors, specifically, misspelled words. We noticed that misspelling is the main reason for some of the observed out-of-vocabulary words. Consequently, we simply fix the unseen misspelled words using Microsoft spell-checker (other approaches like edit distance can also be used). Moreover, while

dealing with an unseen word during evaluation, we try to: 1) replace it with its lower case, or 2) split the word when it contains a “-” (e.g. “marsh-like”) or starts with “un” (e.g. “unloading”). If we still could not find the word in our vocabulary, we consider it as an *unknown* word. In the next subsection, we demonstrate the importance and impact of such trivial preprocessing (see Section B in the Appendix for additional details).

#### 4.5 Results

Table 2 shows the accuracy of the models on training and test sets of SNLI. The first row represents a baseline classifier presented by Bowman et al. (2015) that utilizes handcrafted features. All other listed models are deep-learning based. The gap between the traditional model and deep learning models demonstrates the effectiveness of deep learning methods for this task. We also report the estimated human performance on the SNLI dataset, which is the average accuracy of five annotators in comparison to the gold labels (Gong et al., 2017). It is noteworthy that recent deep learning models surpass the human performance in the NLI task.

As shown in Table 2, previous deep learning models (rows 2-19) can be divided into three categories: 1) sentence encoding based models (rows 2-7), 2) single inter-sentence attention-based models (rows 8-16), and 3) ensemble inter-sentence attention-based models (rows 17-19). We can see that inter-sentence attention-based models perform better than sentence encoding based models, which supports our intuition. Natural language inference requires a deep interaction between the premise and hypothesis. Inter-sentence attention-based approaches can provide such interaction while sentence encoding based models fail to do so.

To further enhance the modeling of interaction between the premise and hypothesis for efficient disambiguation of their relationship, we introduce the dependent reading strategy in our proposed DR-BiLSTM model. The results demonstrate the effectiveness of our model. DR-BiLSTM (Single) achieves 88.5% accuracy on the test set which is noticeably the best reported result among the existing single models for this task. Note that the difference between DR-BiLSTM and Chen et al. (2017) is statistically significant with a p-value of

Model	Accuracy	
	Train	Test
(Bowman et al., 2015) (Feature)	99.7%	78.2%
(Bowman et al., 2015)	83.9%	80.6%
(Vendrov et al., 2015)	98.8%	81.4%
(Mou et al., 2016)	83.3%	82.1%
(Bowman et al., 2016)	89.2%	83.2%
(Liu et al., 2016b)	84.5%	84.2%
(Yu and Munkhdalai, 2017a)	86.2%	84.6%
(Rocktäschel et al., 2015)	85.3%	83.5%
(Wang and Jiang, 2016)	92.0%	86.1%
(Liu et al., 2016a)	88.5%	86.3%
(Parikh et al., 2016)	90.5%	86.8%
(Yu and Munkhdalai, 2017b)	88.5%	87.3%
(Sha et al., 2016)	90.7%	87.5%
(Wang et al., 2017) (Single)	90.9%	87.5%
(Chen et al., 2017) (Single)	92.6%	88.0%
(Gong et al., 2017) (Single)	91.2%	88.0%
(Chen et al., 2017) (Ensemble)	93.5%	88.6%
(Wang et al., 2017) (Ensemble)	93.2%	88.8%
(Gong et al., 2017) (Ensemble)	92.3%	88.9%
Human Performance (Estimated)	97.2%	87.7%
DR-BiLSTM (Single)	94.1%	<b>88.5%</b>
DR-BiLSTM (Single)+Process	94.1%	<b>88.9%</b>
DR-BiLSTM (Ensemble)	94.8%	<b>89.3%</b>
DR-BiLSTM (Ensem.)+Process	94.8%	<b>89.6%</b>

Table 2: Accuracies of the models on the training set and test set of SNLI. DR-BiLSTM (Ensemble) achieves the accuracy of 89.3%, the best result observed on SNLI, while DR-BiLSTM (Single) obtains the accuracy of 88.5%, which considerably outperforms the previous non-ensemble models. Also, utilizing a trivial preprocessing step yields to further improvements of 0.4% and 0.3% for single and ensemble DR-BiLSTM models respectively.

$< 0.001$  over the *Chi-square* test<sup>1</sup>.

To further improve the performance of NLI systems, researchers have built ensemble models. Previously, ensemble systems obtained the best performance on SNLI with a huge margin. Table 2 shows that our proposed single model achieves competitive results compared to these reported ensemble models. Our ensemble model considerably outperforms the current state-of-the-art by obtaining 89.3% accuracy.

Up until this point, we discussed the performance of our models where we have not considered preprocessing for recovering the out-of-vocabulary words. In Table 2, “DR-BiLSTM (Single) + Process”, and “DR-BiLSTM (Ensem.) + Process” represent the performance of our models on the preprocessed dataset. We can see that

<sup>1</sup>Chi-square test ( $\chi^2$  test) is used to determine if there is a significant difference between two categorical variables (i.e. models’ outputs).

our preprocessing mechanism leads to further improvements of 0.4% and 0.3% on the SNLI test set for our single and ensemble models respectively. In fact, our single model (“DR-BiLSTM (Single) + Process”) obtains the state-of-the-art performance over both reported single and ensemble models by performing a simple preprocessing step. Furthermore, “DR-BiLSTM (Ensem.) + Process” outperforms the existing state-of-the-art remarkably (0.7% improvement). For more comparison and analyses, we use “DR-BiLSTM (Single)” and “DR-BiLSTM (Ensemble)” as our single and ensemble models in the rest of the paper.

#### 4.6 Ablation and Configuration Study

We conducted an ablation study on our model to examine the importance and effect of each major component. Then, we study the impact of BiLSTM dimensionality on the performance of the development set and training set of SNLI. We investigate all settings on the development set of the SNLI dataset.

Model	Dev Acc <sup>a</sup>	p-value
DR-BiLSTM	<b>88.69%</b>	-
DR-BiLSTM - hidden MLP	88.45%	<0.001
DR-BiLSTM - average pooling	88.50%	<0.001
DR-BiLSTM - max pooling	88.39%	<0.001
DR-BiLSTM - elem. prd <sup>b</sup>	88.51%	<0.001
DR-BiLSTM - difference	88.24%	<0.001
DR-BiLSTM - diff <sup>c</sup> & elem. prd	87.96%	<0.001
DR-BiLSTM - inference pooling	88.46%	<0.001
DR-BiLSTM - dep. infer <sup>d</sup>	88.43%	<0.001
DR-BiLSTM - dep. enc <sup>e</sup>	88.26%	<0.001
DR-BiLSTM - dep. enc & infer	88.20%	<0.001

<sup>a</sup>Dev Acc, Development Accuracy.

<sup>b</sup>elem. prd, element-wise product.

<sup>c</sup>diff, difference.

<sup>d</sup>dep. infer, dependent reading inference.

<sup>e</sup>dep. enc, dependent reading encoding.

Table 3: Ablation study results. Performance of different configurations of the proposed model on the development set of SNLI along with their p-values in comparison to DR-BiLSTM (Single).

Table 3 shows the ablation study results on the development set of SNLI along with the statistical significance test results in comparison to the proposed model, DR-BiLSTM. We can see that all modifications lead to a new model and their differences are statistically significant with a p-value of < 0.001 over *Chi square* test.

Table 3 shows that removing any part from our model hurts the development set accuracy which indicates the effectiveness of these components.

Among all components, three of them have noticeable influences: max pooling, difference in the attention stage, and dependent reading.

Most importantly, the last four study cases in Table 3 (rows 8-11) verify the main intuitions behind our proposed model. They illustrate the importance of our proposed dependent reading strategy which leads to significant improvement, specifically in the encoding stage. We are convinced that the importance of dependent reading in the encoding stage originates from its ability to focus on more important and relevant aspects of the sentences due to its prior knowledge of the other sentence during the encoding procedure.

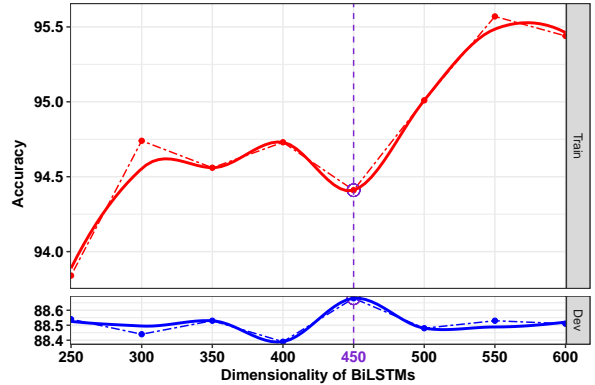


Figure 3: Impact of BiLSTM dimensionality in the proposed model on the training set (red, top) and development set (blue, bottom) accuracies of the SNLI dataset.

Figure 3 shows the behavior of the proposed model accuracy on the training set and development set of SNLI. Since the models are selected based on the best observed development set accuracy during the training procedure, the training accuracy curve (red, top) is not strictly increasing. Figure 3 demonstrates that we achieve the best performance with 450-dimensional BiLSTMs. In other words, using BiLSTMs with lower dimensionality causes the model to suffer from the lack of space for capturing proper information and dependencies. On the other hand, using higher dimensionality leads to overfitting which hurts the performance on the development set. Hence, we use 450-dimensional BiLSTM in our proposed model.

#### 4.7 Analysis

We first investigate the performance of our models categorically. Then, we show a visualization of the

energy function in the attention stage (Equation 3) for an instance from the SNLI test set.

To qualitatively evaluate the performance of our models, we design a set of annotation tags that can be extracted automatically. This design is inspired by the reported annotation tags in Williams et al. (2017). The specifications of our annotation tags are as follows:

- **High Overlap:** premise and hypothesis sentences share more than 70% tokens.
- **Regular Overlap:** sentences share between 30% and 70% tokens.
- **Low Overlap:** sentences share less than 30% tokens.
- **Long Sentence:** either sentence is longer than 20 tokens.
- **Regular Sentence:** premise or hypothesis length is between 5 and 20 tokens.
- **Short Sentence:** either sentence is shorter than 5 tokens.
- **Negation:** negation is present in a sentence.
- **Quantifier:** either of the sentences contains one of the following quantifiers: much, enough, more, most, less, least, no, none, some, any, many, few, several, almost, nearly.
- **Belief:** either of the sentences contains one of the following belief verbs: know, believe, understand, doubt, think, suppose, recognize, forget, remember, imagine, mean, agree, disagree, deny, promise.

Table 4 shows the frequency of aforementioned annotation tags in the SNLI test set along with the performance (accuracy) of ESIM (Chen et al., 2017), DR-BiLSTM (Single), and DR-BiLSTM (Ensemble). Table 4 can be divided into four major categories: 1) gold label data, 2) word overlap, 3) sentence length, and 4) occurrence of special words. We can see that DR-BiLSTM (Ensemble) performs the best in all categories which matches our expectation. Moreover, DR-BiLSTM (Single) performs noticeably better than ESIM in most of the categories except “Entailment”, “High Overlap”, and “Long Sentence”, for which our model is not far behind (gaps of 0.2%, 0.5%, and 0.9%, respectively). It is noteworthy that DR-BiLSTM

Annotation Tag	Freq <sup>a</sup>	ESIM	DR(S) <sup>b</sup>	DR(E) <sup>c</sup>
Entailment	34.3%	90.0%	89.8%	90.9%
Neutral	32.8%	83.7%	85.1%	85.6%
Contradiction	32.9%	90.0%	90.5%	91.4%
High Overlap	24.3%	91.2%	90.7%	92.1%
Reg. Overlap	33.7%	87.1%	87.9%	88.8%
Low Overlap	45.4%	87.0%	87.8%	88.4%
Long Sentence	6.4%	92.2%	91.3%	91.9%
Reg. Sentence	74.9%	87.8%	88.4%	89.2%
Short Sentence	19.9%	87.6%	88.1%	89.3%
Negation	2.1%	82.2%	85.7%	87.1%
Quantifier	8.7%	85.5%	87.4%	87.6%
Belief	0.2%	78.6%	78.6%	78.6%

<sup>a</sup>Freq, Frequency.

<sup>b</sup>DR(S), DR-BiLSTM (Single).

<sup>c</sup>DR(E), DR-BiLSTM (Ensemble).

Table 4: Categorical performance analyses (accuracy) of ESIM (Chen et al., 2017), DR-BiLSTM (DR(S)) and Ensemble DR-BiLSTM (DR(E)) on the SNLI test set.

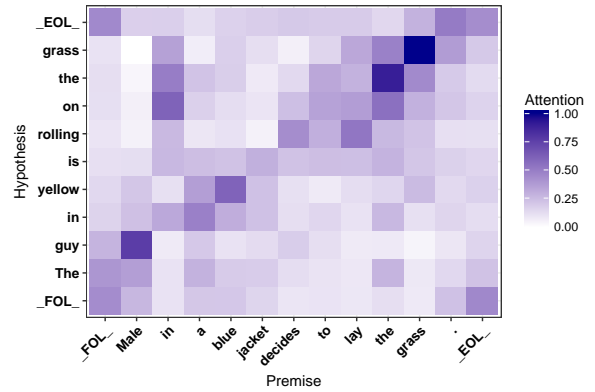


Figure 4: Normalized attention weights for a sample from the SNLI test set. Darker color illustrates higher attention.

(Single) performs better than ESIM in more frequent categories. Specifically, the performance of our model in “Neutral”, “Negation”, and “Quantifier” categories (improvements of 1.4%, 3.5%, and 1.9%, respectively) indicates the superiority of our model in understanding and disambiguating complex samples. Our investigations indicate that ESIM generates somewhat uniform attention for most of the word pairs while our model could effectively attend to specific parts of the given sentences and provide more meaningful attention. In other words, the dependent reading strategy enables our model to achieve meaningful representations, which leads to better attention to obtain further gains on such categories like Negation and Quantifier sentences (see Section C in the Appendix for additional details).



Finally, we show a visualization of the normalized attention weights (energy function, Equation 3) of our model in Figure 4. We show a sentence pair, where the premise is “*Male in a blue jacket decides to lay the grass.*”, and the hypothesis is “*The guy in yellow is rolling on the grass.*”, and its logical relationship is *contradiction*. Figure 4 indicates the model’s ability in attending to critical pairs of words like <Male, guy>, <decides, rolling>, and <lay, rolling>. Finally, high attention between {decides, lay} and {rolling}, and {Male} and {guy} leads the model to correctly classify the sentence pair as *contradiction* (for more samples with attention visualizations, see Section D of the Appendix).

## 5 Conclusion

We propose a novel natural language inference model (DR-BiLSTM) that benefits from a dependent reading strategy and achieves the state-of-the-art results on the SNLI dataset. We also introduce a sophisticated ensemble strategy and illustrate its effectiveness through experimentation. Moreover, we demonstrate the importance of a simple preprocessing step on the performance of our proposed models. Evaluation results show that the preprocessing step allows our DR-BiLSTM (single) model to outperform all previous single and ensemble methods. Similar superior performance is also observed for our DR-BiLSTM (ensemble) model. We show that our ensemble model outperforms the existing state-of-the-art by a considerable margin of 0.7%. Finally, we perform an extensive analysis to demonstrate the strength and weakness of the proposed model, which would pave the way for further improvements in this domain.

## References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. *Neural machine translation by jointly learning to align and translate*. *CoRR* abs/1409.0473. <http://arxiv.org/abs/1409.0473>.
- Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. *A large annotated corpus for learning natural language inference*. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*. pages 632–642. <http://aclweb.org/anthology/D/D15/D15-1075.pdf>.
- Samuel R. Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D. Manning, and Christopher Potts. 2016. *A fast unified model for parsing and sentence understanding*. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. <http://aclweb.org/anthology/P/P16/P16-1139.pdf>.
- Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. *Enhanced LSTM for natural language inference*. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*. pages 1657–1668. <https://doi.org/10.18653/v1/P17-1152>.
- Yichen Gong, Heng Luo, and Jian Zhang. 2017. *Natural language inference over interaction space*. *CoRR* abs/1709.04348. <http://arxiv.org/abs/1709.04348>.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. *Long short-term memory*. *Neural Computation* 9(8):1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Diederik P. Kingma and Jimmy Ba. 2014. *Adam: A method for stochastic optimization*. *CoRR* abs/1412.6980. <http://arxiv.org/abs/1412.6980>.
- Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. 2016. *Ask me anything: Dynamic memory networks for natural language processing*. In *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*. pages 1378–1387. <http://jmlr.org/proceedings/papers/v48/kumar16.html>.
- Pengfei Liu, Xipeng Qiu, Jifan Chen, and Xu-anjing Huang. 2016a. *Deep fusion lstms for text semantic matching*. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. <http://aclweb.org/anthology/P/P16/P16-1098.pdf>.
- Yang Liu, Chengjie Sun, Lei Lin, and Xiaolong Wang. 2016b. *Learning natural language inference using bidirectional LSTM model and inner-attention*. *CoRR* abs/1605.09090. <http://arxiv.org/abs/1605.09090>.
- Bill MacCartney and Christopher D. Manning. 2008. *Modeling semantic containment and exclusion in natural language inference*. In *COLING 2008, 22nd International Conference on Computational Linguistics, Proceedings of the Conference, 18-22 August 2008, Manchester, UK*. pages 521–528. <http://www.aclweb.org/anthology/C08-1066>.
- Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2016. *Natural language inference by tree-based convolution and*

- heuristic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. <http://aclweb.org/anthology/P/P16/P16-2022.pdf>.
- Ankur P. Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2249–2255. <http://aclweb.org/anthology/D/D16/D16-1244.pdf>.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. <http://www.aclweb.org/anthology/D14-1162>.
- Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2641–2649. <https://doi.org/10.1109/ICCV.2015.303>.
- Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kociský, and Phil Blunsom. 2015. Reasoning about entailment with neural attention. *CoRR* abs/1509.06664. <http://arxiv.org/abs/1509.06664>.
- Lei Sha, Baobao Chang, Zhifang Sui, and Sujian Li. 2016. Reading and thinking: Re-read LSTM unit for textual entailment recognition. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 2870–2879. <http://aclweb.org/anthology/C/C16/C16-1270.pdf>.
- Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958. <http://dl.acm.org/citation.cfm?id=2670313>.
- Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2015. Order-embeddings of images and language. *CoRR* abs/1511.06361. <http://arxiv.org/abs/1511.06361>.
- Shuohang Wang and Jing Jiang. 2016. Learning natural language inference with LSTM. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 1442–1451. <http://aclweb.org/anthology/N/N16/N16-1170.pdf>.
- Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 4144–4150. <https://doi.org/10.24963/ijcai.2017/579>.
- Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2017. coverage challenge corpus for sentence understanding through inference. *CoRR* abs/1704.05426. <http://arxiv.org/abs/1704.05426>.
- Hong Yu and Tsendsuren Munkhdalai. 2017a. Neural semantic encoders. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 397–407. <http://aclanthology.info/papers/E17-1038/neural-semantic-encoders>.
- Hong Yu and Tsendsuren Munkhdalai. 2017b. Neural tree indexers for text understanding. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 11–21. <http://aclanthology.info/papers/E17-1002/neural-tree-indexers-for-text-understanding>.
- Kai Zhao, Liang Huang, and Mingbo Ma. 2016. Textual entailment with structured attentions and composition. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 2248–2258. <http://aclweb.org/anthology/C/C16/C16-1212.pdf>.

## A Ensemble Strategy Study

We use the following configurations in our ensemble model study:

- DR-BiLSTM (with different initialization seeds): here, we consider 6 DR-BiLSTMs with different initialization seeds.
- $\tanh$ -Projection: same configuration as DR-BiLSTM, but we use  $\tanh$  instead of  $ReLU$  as the activation function in Equations 6 and 7 in the paper:

$$p_i = \tanh(W_p a_i + b_p) \quad (15)$$

$$q_j = \tanh(W_p b_j + b_p) \quad (16)$$

- DR-BiLSTM (with 1 round of dependent reading): same configuration as DR-BiLSTM, but we do not use dependent reading during the inference process. In other

words, we use  $\tilde{p} = \bar{p}$  and  $\tilde{q} = \bar{q}$  instead of Equations 10 and 11 in the paper respectively.

- DR-BiLSTM (with 3 rounds of dependent reading): same configuration as the above, but we use 3 rounds of dependent reading. Formally, we replace Equations 1 and 2 in the paper with the following equations respectively:

$$\begin{aligned}
 -, s_v &= \text{BiLSTM}(v, 0) \\
 -, s_{vu} &= \text{BiLSTM}(u, s_v) \\
 -, s_{vuv} &= \text{BiLSTM}(v, s_{vu}) \\
 \hat{u}, - &= \text{BiLSTM}(u, s_{vuv})
 \end{aligned} \tag{17}$$

$$\begin{aligned}
 -, s_u &= \text{BiLSTM}(u, 0) \\
 -, s_{uv} &= \text{BiLSTM}(v, s_u) \\
 -, s_{uvu} &= \text{BiLSTM}(u, s_{uv}) \\
 \hat{v}, - &= \text{BiLSTM}(v, s_{uvu})
 \end{aligned} \tag{18}$$

Our final ensemble model, DR-BiLSTM (Ensemble) is the combination of the following 6 models: tanh-Projection, DR-BiLSTM (with 1 round of dependent reading), DR-BiLSTM (with 3 rounds of dependent reading), and 3 DR-BiLSTMs with different initialization seeds.

We also experiment with majority voting and averaging the probability distribution strategies for ensemble models using the same set of models as our weighted averaging ensemble method (as described above). Figure 5 shows the behavior of the majority voting strategy with different number of models. Interestingly, the best development accuracy is also observed using 6 individual models including tanh-Projection, DR-BiLSTM (with 1 round of dependent reading), DR-BiLSTM (with 3 rounds of dependent reading), and 3 DR-BiLSTMs with varying initialization seeds that are different from our DR-BiLSTM (Ensemble) model.

We should note that our weighted averaging ensemble strategy performs better than the majority voting method in both development set and test set of SNLI, which indicates the effectiveness of our approach. Furthermore, our method could show more consistent behavior for training and test sets when we increased the number of models (see Figure 2 in Section 4.3 of the paper). According to

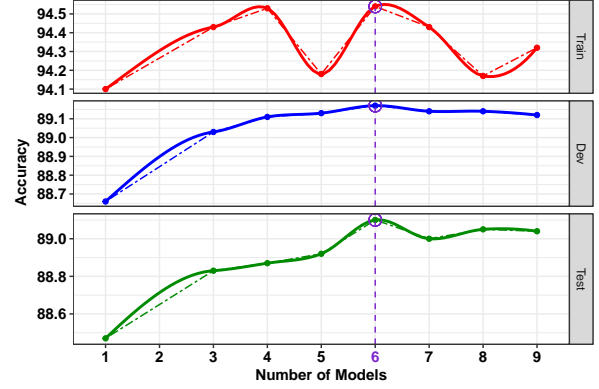


Figure 5: Performance of  $n$  ensemble models using majority voting on natural language inference reported for training set (red, top), development set (blue, middle), and test set (green, bottom) of SNLI. The best performance on development set is used as the criteria to determine the final ensemble. The best performance on development set is observed using 6 models.

our observations, averaging the probability distributions fails to improve the development set accuracy using two and three models, so we did not study it further.

## B Preprocessing Study

Table 5 shows some erroneous sentences from the SNLI test set along with their corrected equivalents (after preprocessing). Furthermore, we show the energy function (Equation 3 in the paper) visualizations of 6 examples from the aforementioned data samples in Figures 6, 7, 8, 9, 10, and 11. Each figure presents the visualization of an original erroneous sample along its corrected version. These figures clearly illustrate that fixing the erroneous words leads to producing correct attentions over the sentences. This can be observed by comparing the attention for the erroneous words and corrected words, e.g. “daschunds” and “dachshunds” in the premise of Figures 6 and 7. Note that we add two dummy notations (i.e. `_FOL_`, and `_EOL_`) to all sentences which indicate their beginning and end.

## C Category Study

Here we investigate the normalized attention weights of DR-BiLSTM and ESIM for four samples that belong to Negation and/or Quantifier categories (Figures 12 - 15). Each figure illustrates the normalized energy function of DR-BiLSTM (left diagram) and ESIM (right diagram) respec-

Original Sentence	Corrected Sentence
<i>Froends</i> ride in an open top vehicle together.	<i>Friends</i> ride in an open top vehicle together.
A middle <i>easten</i> store.	A middle <i>eastern</i> store.
A woman is looking at a <i>photographer</i>	A woman is looking at a <i>photographer</i>
The mother and daughter are <i>fighitn</i> .	The mother and daughter are <i>fighting</i> .
Two <i>kiled</i> men hold bagpipes	Two <i>killed</i> men hold bagpipes
A woman escapes a from a hostile <i>enviroment</i>	A woman escapes a from a hostile <i>environment</i>
Two <i>daschunds</i> play with a red ball	Two <i>dachshunds</i> play with a red ball
A black dog is running through a <i>marsh-like</i> area.	A black dog is running through a <i>marsh like</i> area.
a singer wearing a <i>jacker</i> performs on stage	a singer wearing a <i>jacket</i> performs on stage
There is a <i>sculture</i>	There is a <i>sculpture</i>
Taking a <i>neverending</i> break	Taking a <i>never ending</i> break
The woman has sounds <i>emanting</i> from her mouth.	The woman has sounds <i>emanating</i> from her mouth.
the lady is <i>shipping</i>	the lady is <i>shopping</i>
A Bugatti and a <i>Lambourgini</i> compete in a road race.	A Bugatti and a <i>Lamborghini</i> compete in a road race.

Table 5: Examples of original sentences that contain erroneous words (misspelled) in the test set of SNLI along with their corrected counterparts. Erroneous words are shown in ***bold and italic***.

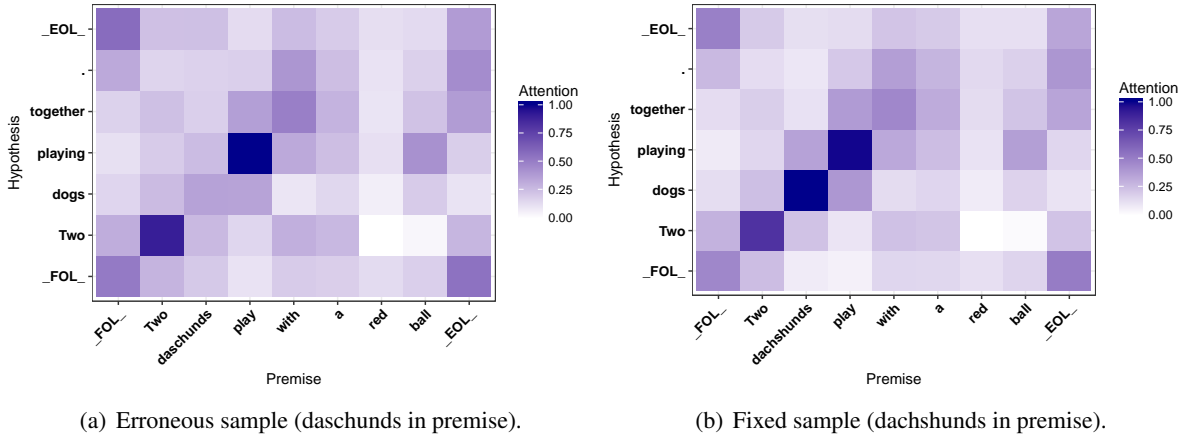


Figure 6: Visualization of the energy function for one erroneous sample (a) and the fixed sample (b). The gold label is *Entailment*. Our model returns *Contradiction* for the erroneous sample, but correctly classifies the fixed sample.

tively. Provided figures indicate that ESIM assigns somewhat similar attention to most of the pairs while DR-BiLSTM focuses on specific parts of the given premise and hypothesis.

## D Attention Study

In this section, we show visualizations of 18 samples of normalized attention weights (energy function, see Equation 3 in the paper). Each column in Figures 16, 17, and 18, represents three data samples that share the same premise but differ in hypothesis. Also, each row is allocated to a specific logical relationship (Top: Entailment, Middle: Neutral, and Bottom: Contradiction). DR-BiLSTM classifies all data samples reported in these figures correctly.



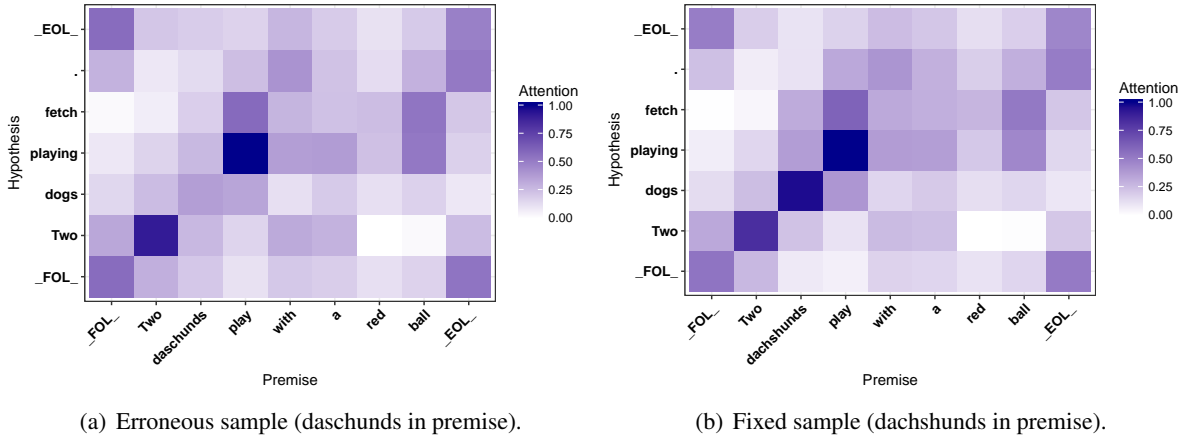


Figure 7: Visualization of the energy function for one erroneous sample (a) and the fixed sample (b). The gold label is *Neutral*. Our model returns *Contradiction* for the erroneous sample, but correctly classifies the fixed sample.

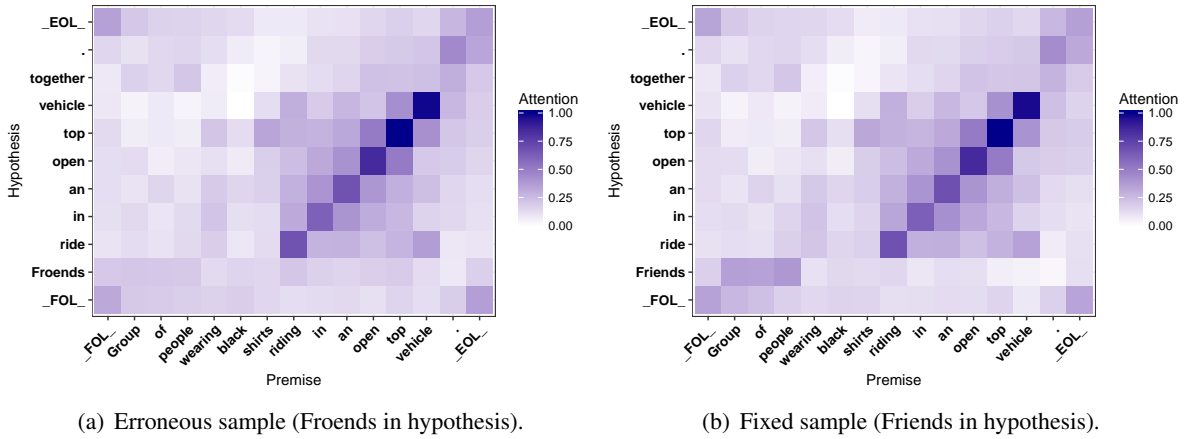


Figure 8: Visualization of the energy function for one erroneous sample (a) and the fixed sample (b). The gold label is *Neutral*. Our model returns *Entailment* for the erroneous sample, but correctly classifies the fixed sample.

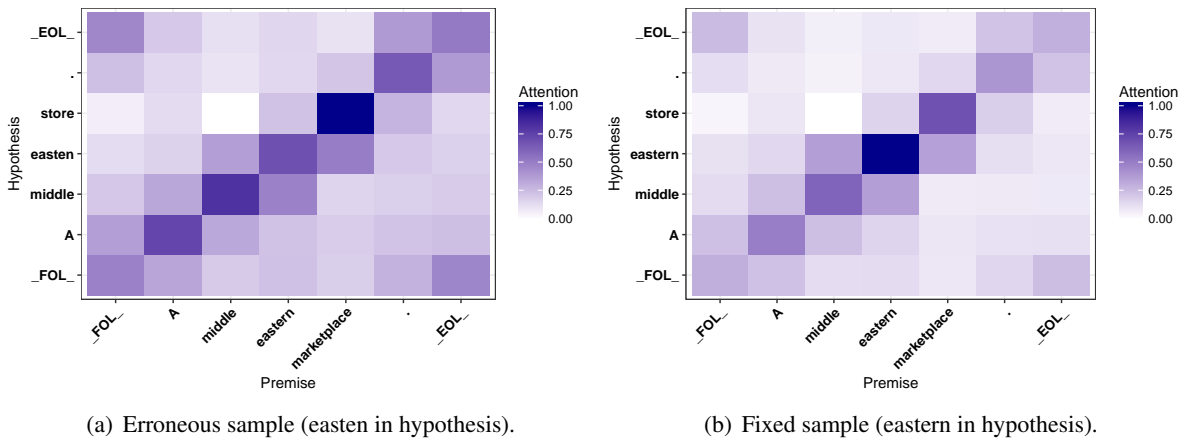


Figure 9: Visualization of the energy function for one erroneous sample (a) and the fixed sample (b). The gold label is *Entailment*. Our model returns *Contradiction* for the erroneous sample, but correctly classifies the fixed sample.

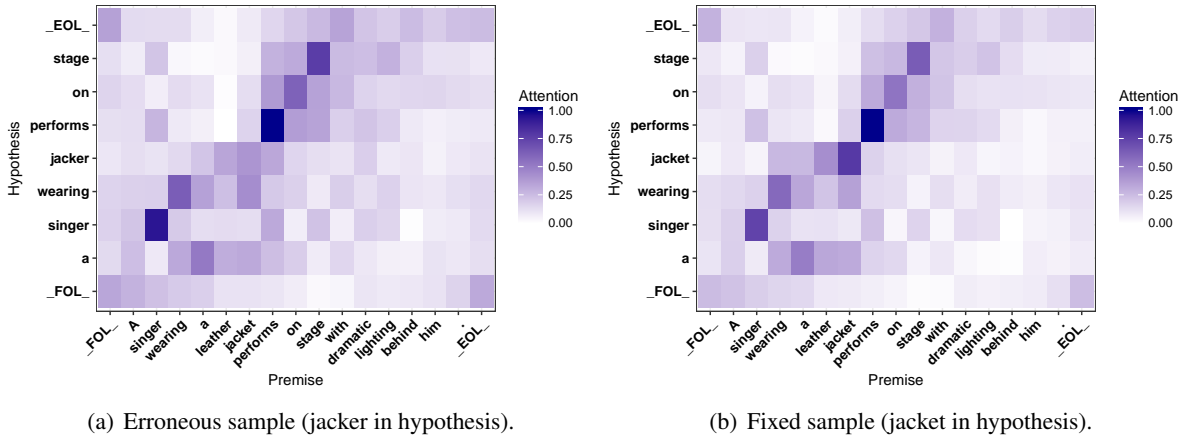


Figure 10: Visualization of the energy function for one erroneous sample (a) and the fixed sample (b). The gold label is *Entailment*. Our model returns *Neutral* for the erroneous sample, but correctly classifies the fixed sample.

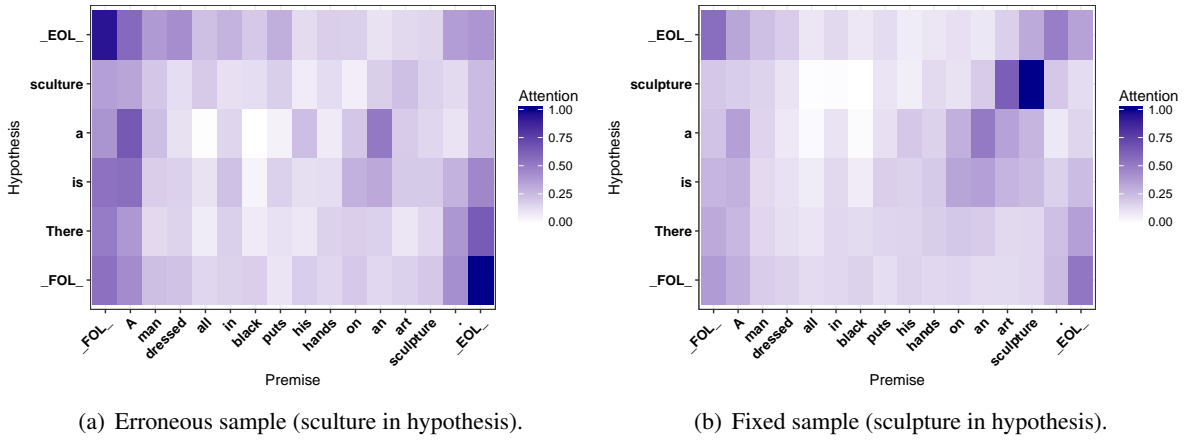


Figure 11: Visualization of the energy function for one erroneous sample (a) and the fixed sample (b). The gold label is *Entailment*. Our model returns *Neutral* for the erroneous sample, but correctly classifies the fixed sample.

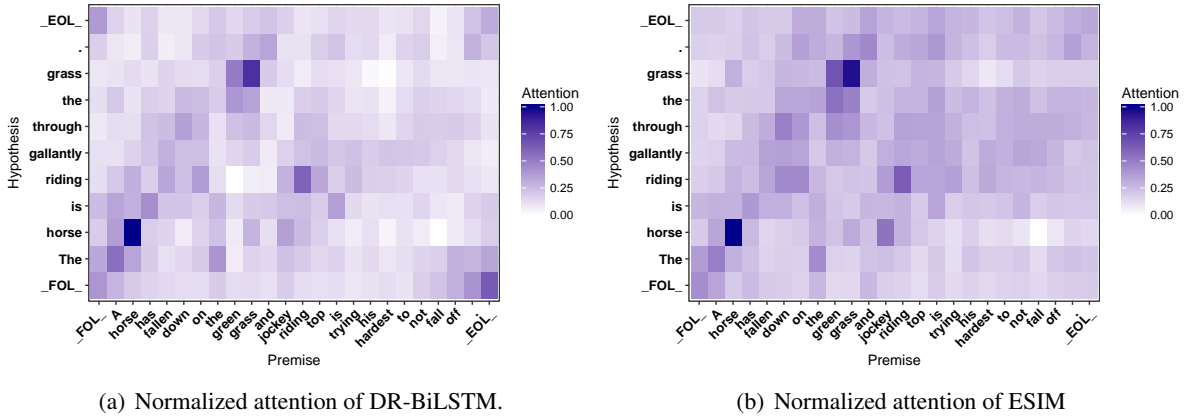
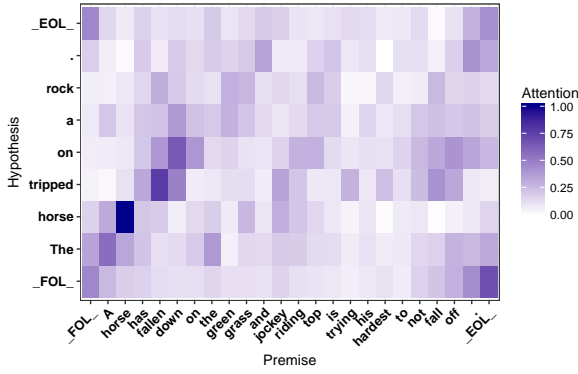
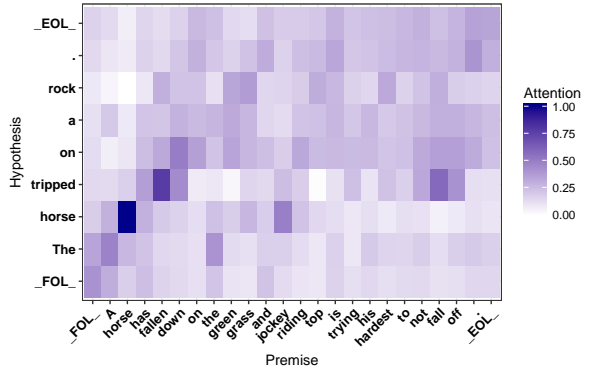


Figure 12: Visualization of the normalized attention weights of DR-BiLSTM (a) and ESIM (b) models for one sample from the SNLI test set. This sample belongs to the Negation category. The gold label is *Contradiction*. Our model returns *Contradiction* while ESIM returns *Entailment*.

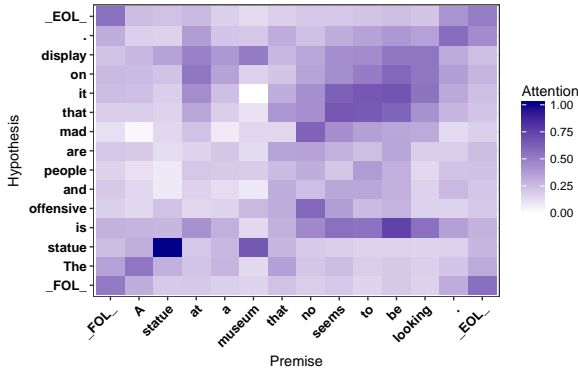


(a) Normalized attention of DR-BiLSTM.

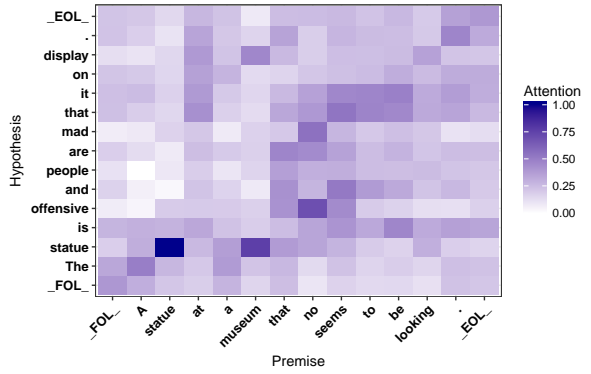


(b) Normalized attention of ESIM

Figure 13: Visualization of the normalized attention weights of DR-BiLSTM (a) and ESIM (b) models for one sample from the SNLI test set. This sample belongs to the Negation category. The gold label is *Contradiction*. Our model returns *Contradiction* while ESIM returns *Entailment*.

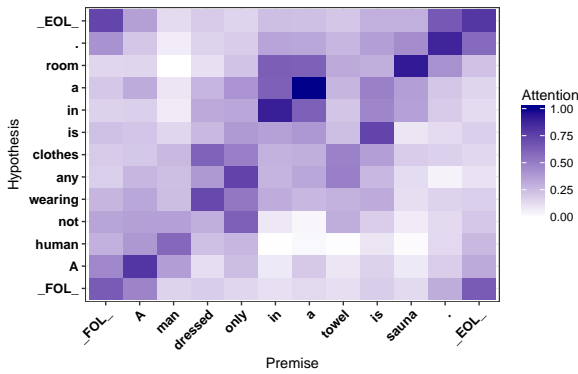


(a) Normalized attention of DR-BiLSTM.

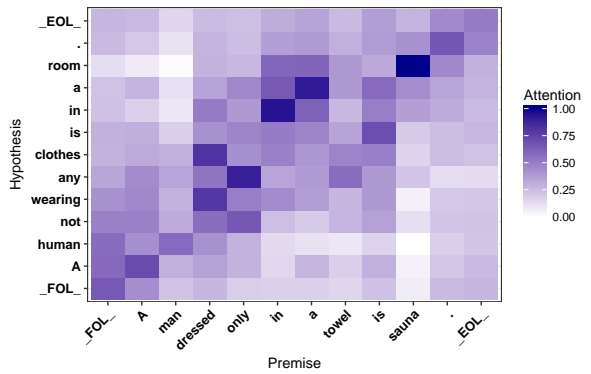


(b) Normalized attention of ESIM

Figure 14: Visualization of the normalized attention weights of DR-BiLSTM (a) and ESIM (b) models for one sample from the SNLI test set. This sample belongs to both Negation and Quantifier categories. The gold label is *Neutral*. Our model returns *Neutral* while ESIM returns *Contradiction*.



(a) Normalized attention of DR-BiLSTM.



(b) Normalized attention of ESIM

Figure 15: Visualization of the normalized attention weights of DR-BiLSTM (a) and ESIM (b) models for one sample from the SNLI test set. This sample belongs to both Negation and Quantifier categories. The gold label is *Entailment*. Our model returns *Entailment* while ESIM returns *Contradiction*.

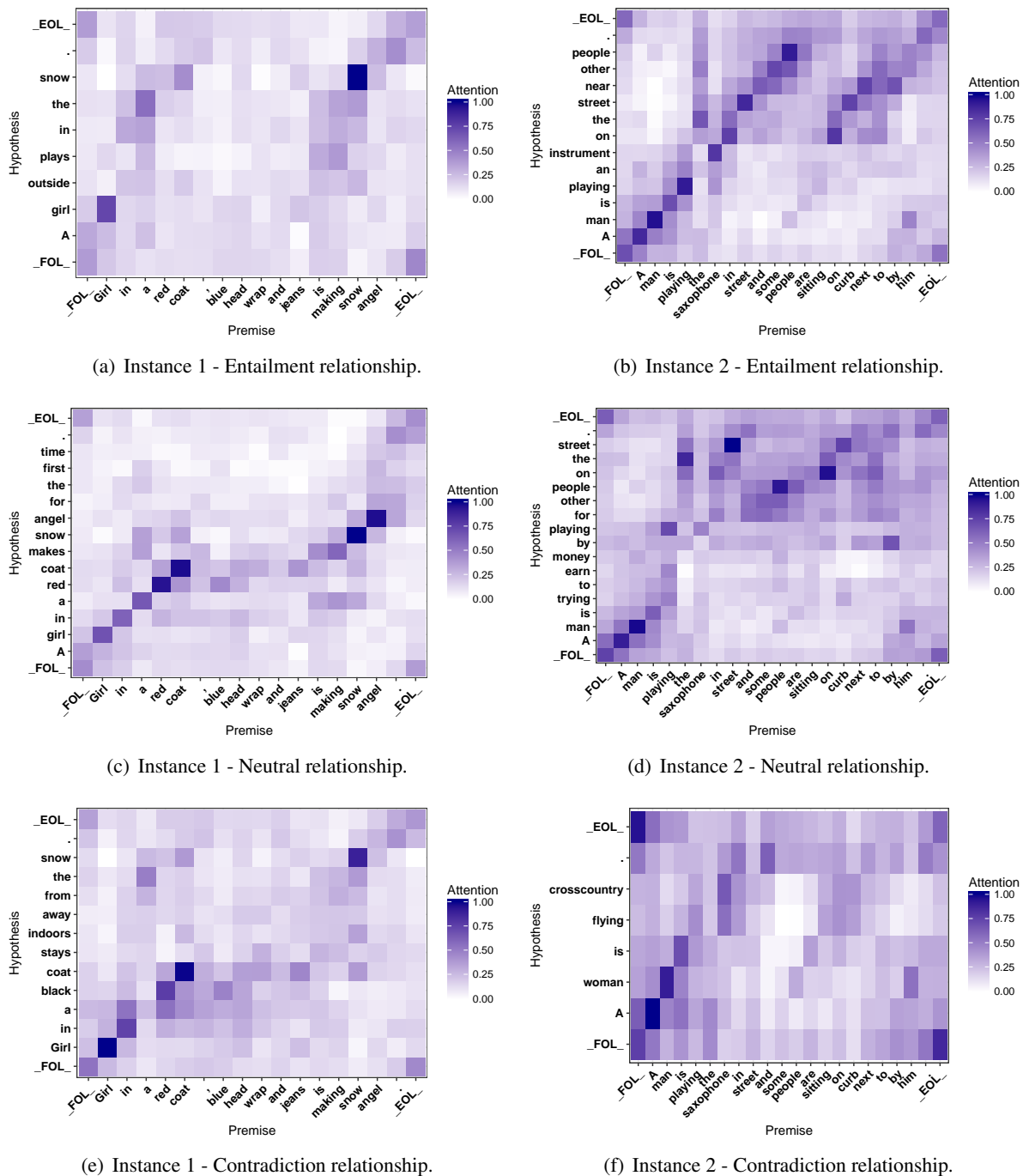
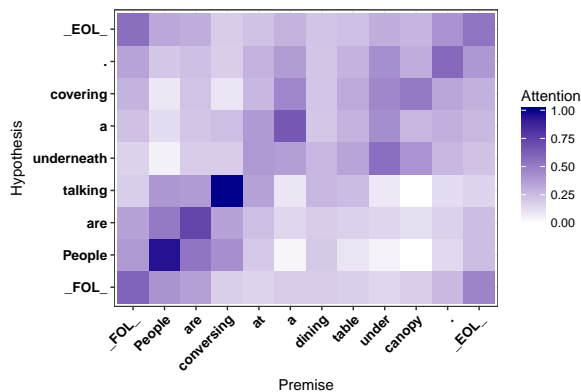
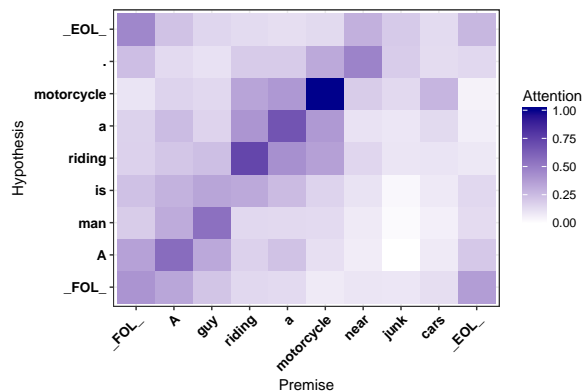


Figure 16: Normalized attention weights for 6 data samples from the test set of SNLI dataset. (a,c,e) and (b,d,f) represent the normalized attention weights for *Entailment*, *Neutral*, and *Contradiction* logical relationships of two premises (Instance 1 and 2) respectively. Darker color illustrates higher attention.

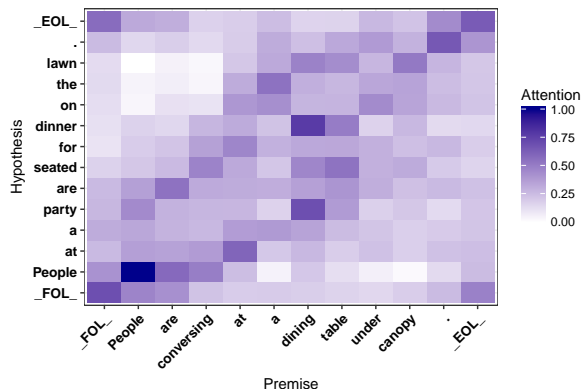




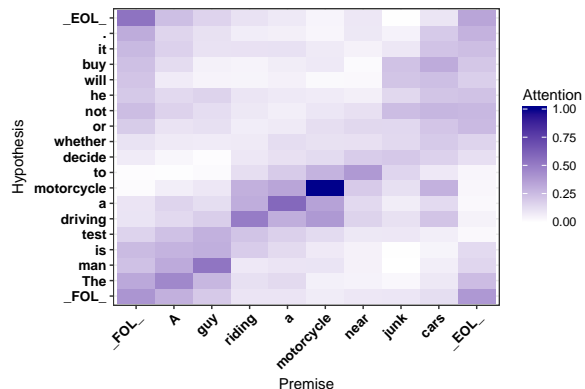
(a) Instance 3 - Entailment relationship.



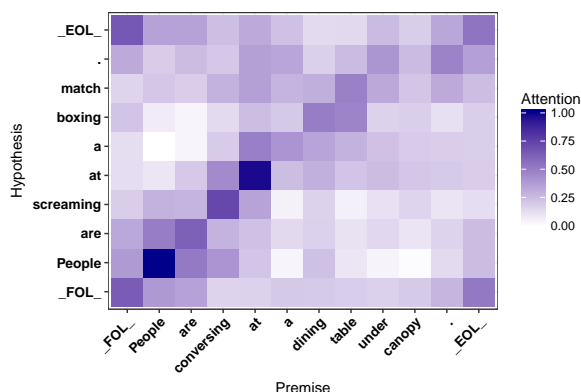
(b) Instance 4 - Entailment relationship.



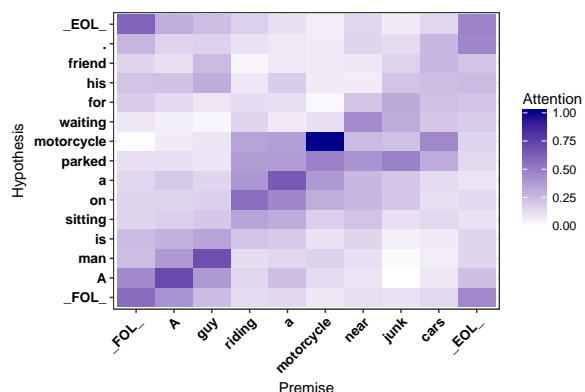
(c) Instance 3 - Neutral relationship.



(d) Instance 4 - Neutral relationship.

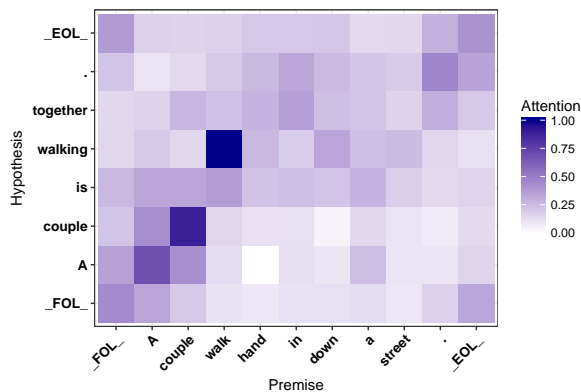


(e) Instance 3 - Contradiction relationship.

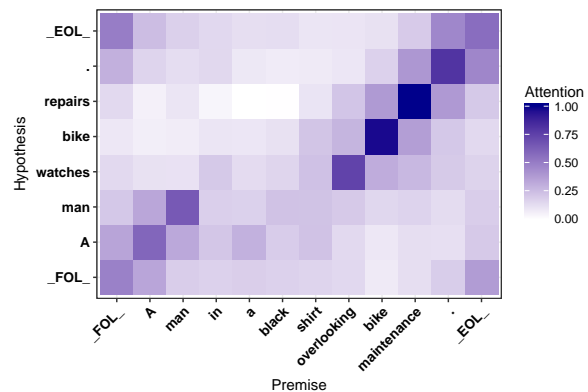


(f) Instance 4 - Contradiction relationship.

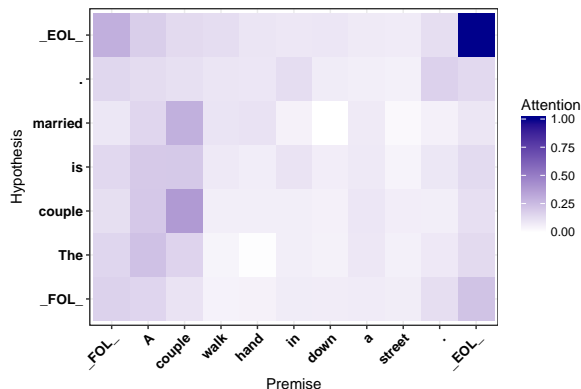
Figure 17: Normalized attention weights for 6 data samples from the test set of SNLI dataset. (a,c,e) and (b,d,f) represent the normalized attention weights for *Entailment*, *Neutral*, and *Contradiction* logical relationships of two premises (Instance 3 and 4) respectively. Darker color illustrates higher attention.



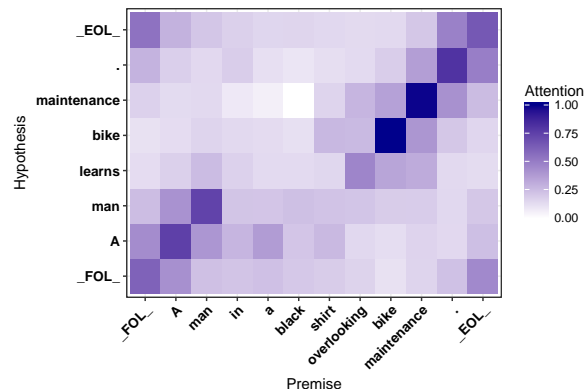
(a) Instance 5 - Entailment relationship.



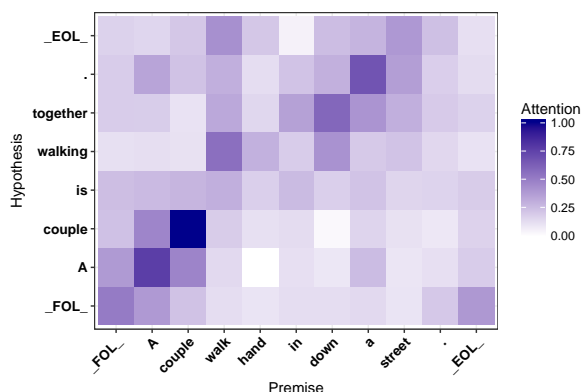
(b) Instance 6 - Entailment relationship.



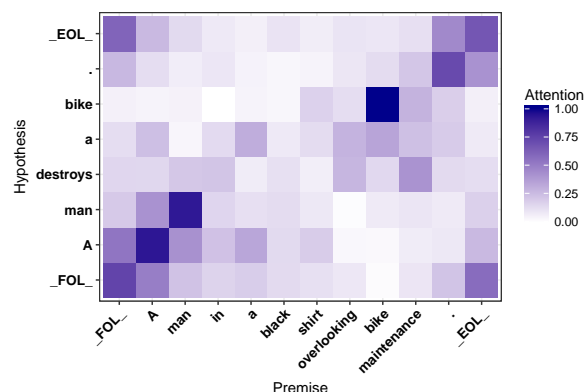
(c) Instance 5 - Neutral relationship.



(d) Instance 6 - Neutral relationship.



(e) Instance 5 - Contradiction relationship.



(f) Instance 6 - Contradiction relationship.

Figure 18: Normalized attention weights for 6 data samples from the test set of SNLI dataset. (a,c,e) and (b,d,f) represent the normalized attention weights for *Entailment*, *Neutral*, and *Contradiction* logical relationships of two premises (Instance 5 and 6) respectively. Darker color illustrates higher attention.