# Learning Natural Language Inference using Bidirectional LSTM model and Inner-Attention

**Yang Liu, Chengjie Sun, Lei Lin** and **Xiaolong Wang**

Harbin Institute of Technology, Harbin, P.R.China

{yliu,cjsun,linl,wangxl}@insun.hit.edu.cn

## Abstract

In this paper, we proposed a sentence encoding-based model for recognizing text entailment. In our approach, the encoding of sentence is a two-stage process. Firstly, average pooling was used over word-level bidirectional LSTM (biLSTM) to generate a first-stage sentence representation. Secondly, attention mechanism was employed to replace average pooling on the same sentence for better representations. Instead of using target sentence to attend words in source sentence, we utilized the sentence's first-stage representation to attend words appeared in itself, which is called "Inner-Attention" in our paper . Experiments conducted on Stanford Natural Language Inference (SNLI) Corpus has proved the effectiveness of "Inner-Attention" mechanism. With less number of parameters, our model outperformed the existing best sentence encoding-based approach by a large margin.

## 1 Introduction

Given a pair of sentences, the goal of recognizing text entailment (RTE) is to determine whether the hypothesis can reasonably be inferred from the premises. There were three types of relation in RTE, Entailment (inferred to be true), Contradiction (inferred to be false) and Neutral (truth unknown).A few examples were given in Table 1.

Traditional methods to RTE has been the dominion of classifiers employing hand engineered features, which heavily relied on natural language processing pipelines and external resources. Formal reasoning methods (Bos and Markert, 2005) were

| P | The boy is running through a grassy area. | |
|---|---|---|
| **H** | The boy is in his room. | C |
| | A boy is running outside. | E |
| | The boy is in a park. | N |

**Table 1:** Examples of three types of label in RTE, where P stands for Premises and H stands for Hypothesis

also explored by many researchers, but not been widely used because of its complexity and domain limitations.

Recently published Stanford Natural Language Inference (SNLI[1]) corpus makes it possible to use deep learning methods to solve RTE problems. So far proposed deep learning approaches can be roughly categorized into two groups: sentence encoding-based models and matching encoding-based models. As the name implies, the encoding of sentence is the core of former methods, while the latter methods directly model the relation between two sentences and didn't generate sentence representations at all.

In view of universality, we focused our efforts on sentence encoding-based model. Existing methods of this kind including: LSTMs-based model, GRUs-based model, TBCNN-based model and SPINN-based model. Single directional LSTMs and GRUs suffer a weakness of not utilizing the contextual information from the future tokens and Convolutional Neural Networks didn't make full use of information contained in word order. Bidirectional LSTM utilizes both the previous and future context by processing the sequence on two directions which helps to address the drawbacks mentioned above.
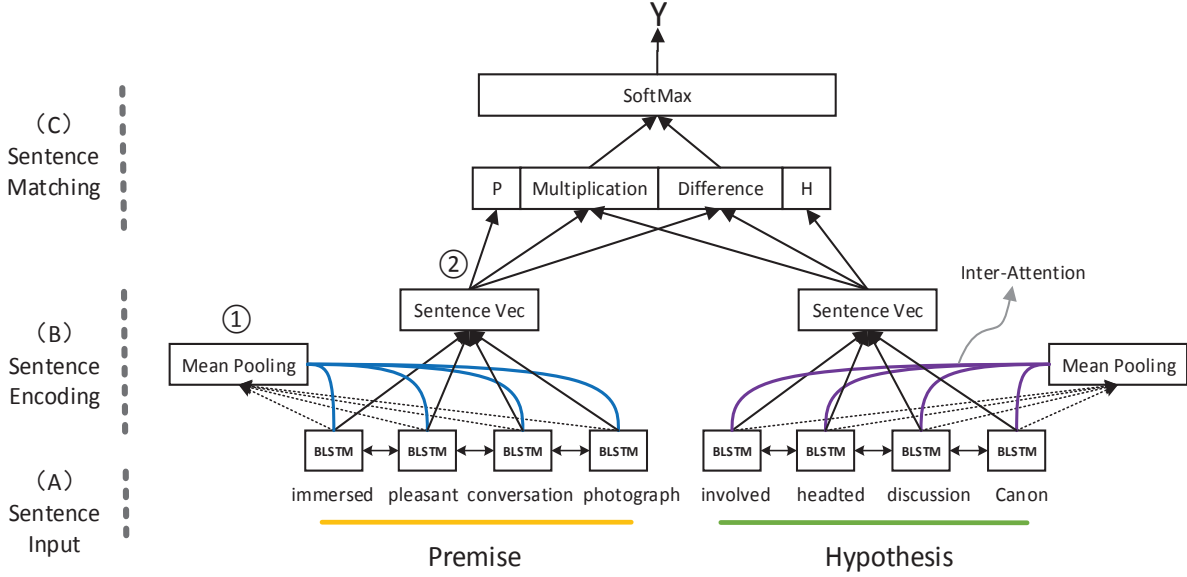
---

[1]http://nlp.stanford.edu/projects/snli/

**Figure 1:** Architecture of Bidirectional LSTM model with Inner-Attention

(Tan et al., 2015)

A recent work by (Rocktäschel et al., 2015) improved the performance by applying a neural attention model that didn't yield sentence embeddings.

In this paper, we proposed a unified deep learning framework for recognizing textual entailment which dose not require any feature engineering, or external resources. The basic model is based on building biLSTM models on both premises and hypothesis. The basic mean pooling encoder can roughly form a intuition about what this sentence is talking about. Obtained this representation, we extended this model by utilize an Inner-Attention mechanism on both sides. This mechanism helps generate more accurate and focused sentence representations for classification. In addition, we introduced a simple effective input strategy that get ride of same words in hypothesis and premise, which further boosts our performance. Without parameter tuning, we improved the art-of-the-state performance of sentence encoding-based model by nearly 2%.

## 2 Our approach

In our work, we treated RTE task as a supervised three-way classification problem. The overall architecture of our model is shown in Figure 1. The design of this model we follow the idea of Siamese Network, that the two identical sentence encoders

share the same set of weights during training, and the two sentence representations then combined together to generated a "relation vector" for classification. As we can see from the figure, the model mainly consists of three parts. From top to bottom were: (A). The sentence input module; (B). The sentence encoding module; (C). The sentence matching module. We will explain the last two parts in detail in the following subsection. And the sentence input module will be introduced in Section 3.3.

### 2.1 Sentence Encoding Module

Sentence encoding module is the fundamental part of this model. To generate better sentence representations, we employed a two-step strategy to encode sentences. Firstly, average pooling layer was built on top of word-level biLSTMs to produce sentence vector. This simple encoder combined with the sentence matching module formed the basic architecture of our model. With much less parameters, this basic model alone can outperformed art-of-state method by a small margin. (refer to Table 3). Secondly, attention mechanism was employed on the same sentence, instead of using target sentence representation to attend words in source sentence, we used the representation generated in previous stage to attend words appeared in the sentence itself, which results in a similar distribution with other

attention mechanism weights. More attention was given to important words.[2]

The idea of "Inner-attention" was inspired by the observation that when human read one sentence, people usually can roughly form an intuition about which part of the sentence is more important according past experience. And we implemented this idea using attention mechanism in our model. The attention mechanism is formalized as follows:

$$M = tanh(W^y Y + W^h R_{ave} \otimes e_L)$$
$$\alpha = softmax(w^T M)$$
$$R_{att} = Y\alpha^T$$

where $Y$ is a matrix consisting of output vectors of biLSTM, $R_{ave}$ is the output of mean pooling layer, $\alpha$ denoted the attention vector and $R_{att}$ is the *attention-weighted* sentence representation.

## 2.2 Sentence Matching Module

Once the sentence vectors are generated. Three matching methods were applied to extract relations between premise and hypothesis.

- Concatenation of the two representations
- Element-wise product
- Element-wise difference

This matching architecture was first used by (Mou et al., 2015). Finally, we used a SoftMax layer over the output of a non-linear projection of the generated matching vector for classification.

## 3 Experiments

### 3.1 DataSet

To evaluate the performance of our model, we conducted our experiments on Stanford Natural Language Inference (SNLI) corpus (Bos and Markert, 2005). At 570K pairs, SNLI is two orders of magnitude larger than all other resources of its type. The dataset is constructed by crowdsourced efforts, each sentence written by humans. The target labels comprise three classes: Entailment, Contradiction, and Neutral

(two irrelevant sentences). We applied the standard train/validation/test split, containing 550k, 10k, and 10k samples, respectively.

### 3.2 Parameter Setting

The training objective of our model is cross-entropy loss, and we use minibatch SGD with the Rmsprop (Tieleman and Hinton, 2012) for optimization. The batch size is 128. A dropout layer was applied in the output of the network with the dropout rate set to 0.25. In our model, we used pretrained 300D Glove 840B vectors (Pennington et al., 2014) to initialize the word embedding. Out-of-vocabulary words in the training set are randomly initialized by sampling values uniformly from (0.05, 0.05). All of these embedding are not updated during training . We didn't tune representations of words for two reasons: 1. To reduced the number of parameters needed to train. 2. Keep their representation stays close to unseen similar words in inference time, which improved the model's generation ability. The model is implemented using open-source framework Keras.[3]

### 3.3 The Input Strategy

In this part, we investigated four strategies to modify the input on our basic model which helps us increase performance, the four strategies are:

- Inverting Premises (Sutskever et al., 2014)
- Doubling Premises (Zaremba and Sutskever, 2014)
- Doubling Hypothesis
- Differentiating Inputs (Removing same words appeared in premises and hypothesis)

Experimental results were illustrated in Table 2. As we can see from it, doubling hypothesis and differentiating inputs both improved our model's performance.While the hypothesises usually much shorter than premises, doubling hypothesis may absorb this difference and emphasize the meaning twice via this strategy. Differentiating input strategy forces the model to focus on different part of the two sentences which may help the classification for Neutral and Contradiction examples as we observed that our model tended to assign unconfident instances to Entailment. And the original input sentences appeared in Figure 1 are:

**Premise:** *Two man in polo shirts and tan pants immersed in a pleasant conversation about photograph.*

---

[2]Recently, (Yang et al., 2016) proposed a Hierarchical Attention model on the task of document classification also used for but the target representation in attention their mechanism is randomly initialized.

[3]http://keras.io/

| Input Strategy | Test Acc. |
|---|---|
| Original Sequences | 83.24% |
| Inverting Premises | 82.60% |
| Doubling Premises | 83.66% |
| Doubling Hypothesis | 82.83% |
| Differentiating Inputs | **83.72%** |

**Table 2:** Comparison of different input strategies

**Hypothesis:** *Two man in polo shirts and tan pants involved in a heated discussion about Canon.*
**Label:** *Contradiction*

While most of the words in this pair of sentences are same or close in semantic, It is hard for model to distinguish the difference between them, which resulted in labeling it with Neutral or Entailment. Through differentiating inputs strategy, this kind of problems can be solved.

### 3.4 Comparison Methods

In this part, we compared our model against the following art-of-the-state baseline approaches:

- **LSTM enc:** 100D LSTM encoders + MLP. (Bowman et al., 2015)
- **GRU enc:** 1024D GRU encoders + skip-thoughts + cat, -. (Vendrov et al., 2015)
- **TBCNN enc:** 300D Tree-based CNN encoders + cat, ○ , -. (Mou et al., 2015)
- **SPINN enc:** 300D SPINN-NP encoders + cat, ○ , -. (Bowman et al., 2016)
- **Static-Attention:** 100D LSTM + static attention. (Rocktäschel et al., 2015)
- **WbW-Attention:** 100D LSTM + word-by-word attention. (Rocktäschel et al., 2015)

The *cat* refers to concatenation, **-** and ○ denote element-wise difference and product, respectively. Much simpler and easy to understand.

### 3.5 Results and Qualitative Analysis

Although the classification of RTE example is not solely relying on representations obtained from attention, it is still instructive to analysis Inner-Attention mechanism as we witnessed a large performance increase after employing it. We hand-picked several examples from the dataset to visualize. In order to make the weights more discriminated, we didn't use a uniform colour atla cross sentences. That is, each sentence have its own color atla, the lightest color and the darkest color denoted the smallest attention weight the biggest value

| Model | Params | Test Acc. |
|---|---|---|
| *Sentence encoding-based models* | | |
| LSTM enc | 3.0M | 80.6% |
| GRU enc | 15M | 81.4% |
| TBCNN enc | 3.5M | 82.1% |
| SPINN enc | 3.7M | 83.2% |
| Basic model | **2.0M** | 83.3% |
| *+ Inner-Attention* | 2.8M | 84.2% |
| *+ Diversing Input* | 2.8M | **85.0%** |
| *Other neural network models* | | |
| Static-Attention | 242K | 82.4% |
| WbW-Attention | 252K | 83.5% |

**Table 3:** Performance comparison of different models on SNLI.

within the sentence, respectively. Visualizations of Inner-Attention on these examples are depicted in Figure 2.
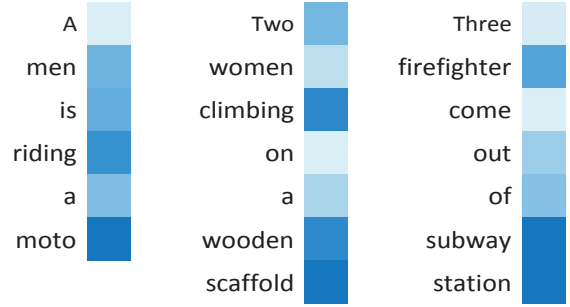


**Figure 2:** Inner-Attention Visualizations.

We observed that more attention was given to Nones, Verbs and Adjectives. This conform to our experience that these words are more semantic richer than function words. While mean pooling regarded each word of equal importance, the attention mechanism helps re-weight words according to their importance. And more focused and accurate sentence representations were generated based on produced attention vectors.

## 4 Conclusion and Future work

In this paper, we proposed a bidirectional LSTM-based model with Inner-Attention to solve the RTE problem. We come up with an idea to utilize attention mechanism within sentence which can teach itself to attend words without the information from another one. The Inner-Attention mechanism helps produce more accurate sentence representa-

tions through attention vectors. In addition, the simple effective diversing input strategy introduced by us further boosts our results. And this model can be easily adapted to other sentence-matching models. Our future work including:

1. Employ this architecture on other sentence-matching tasks such as Question Answer, Paraphrase and Sentence Text Similarity etc.
2. Try more heuristics matching methods to make full use of the sentence vectors.

## Acknowledgments

## References

[Bos and Markert2005] Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 628–635. Association for Computational Linguistics.

[Bowman et al.2015] Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

[Bowman et al.2016] 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. *arXiv preprint arXiv:1603.06021*.

[Mou et al.2015] Lili Mou, Men Rui, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2015. Recognizing entailment and contradiction by tree-based convolution. *arXiv preprint arXiv:1512.08422*.

[Pennington et al.2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

[Rocktäschel et al.2015] Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiský, and Phil Blunsom. 2015. Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

[Sutskever et al.2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

[Tan et al.2015] Ming Tan, Bing Xiang, and Bowen Zhou. 2015. Lstm-based deep learning models for non-factoid answer selection. *arXiv preprint arXiv:1511.04108*.

[Tieleman and Hinton2012] Tijmen Tieleman and Geoffrey Hinton. 2012. Lecture 6.5-rmsprop. *COURSERA: Neural networks for machine learning*.

[Vendrov et al.2015] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2015. Order-embeddings of images and language. *arXiv preprint arXiv:1511.06361*.

[Yang et al.2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

[Zaremba and Sutskever2014] Wojciech Zaremba and Ilya Sutskever. 2014. Learning to execute. *arXiv preprint arXiv:1410.4615*.