Generating Textual Entailment Using Residual LSTMs

Maosheng Guo([0000-0002-3829-1179] ✉), Yu Zhang, Dezhi Zhao, and Ting Liu

School of Computer Science and Technology,
Harbin Institute of Technology, Harbin 150001, China
{msguo, zhangyu, dzzhao, tliu}@ir.hit.edu.cn

Abstract. Generating textual entailment (GTE) is a recently proposed task to study how to infer a sentence from a given premise. Current sequence-to-sequence GTE models are prone to produce invalid sentences when facing with complex enough premises. Moreover, the lack of appropriate evaluation criteria hinders researches on GTE. In this paper, we conjecture that the unpowerful encoder is the major bottleneck in generating more meaningful sequences, and improve this by employing the residual LSTM network. With the extended model, we obtain state-of-the-art results. Furthermore, we propose a novel metric for GTE, namely EBR (Evaluated By Recognizing textual entailment), which could evaluate different GTE approaches in an objective and fair way without human effort while also considering the diversity of inferences. In the end, we point out the limitation of adapting a general sequence-to-sequence framework under GTE settings, with some proposals for future research, hoping to generate more public discussion.

Keywords: Generating Textual Entailment, Natural Language Generation, Natural Language Processing, Artificial Intelligence.

1 Introduction

The ability of reasoning in natural language is necessary for many information access applications such as question-answering systems, where the answer to a question should be inferred from the supporting text. The reasoning relationship in texts is defined as textual entailment [4]. There are two major tasks to study this relationship: Recognizing Textual Entailment (RTE) and Generating Textual Entailment (GTE).

Compared with RTE which is well studied, GTE is a rather new task which was formally proposed very recently to overcome the shortcomings when applying RTE techniques in downstream NLP tasks, such as question answering and text summarization, where only one source sentence is available and models need to come up with their own hypotheses by inference according to commonsense knowledge [7].

In GTE settings, the system is asked to produce a new sentence called hypothesis (e.g. S2 in **Fig. 1**) according to a given text known as a premise (e.g. S1). Rule-based algorithms [6,9] and sequence-to-sequence LSTM model [7] were proposed to generate inferences.

S1. A group of people prepare hot air balloons for takeoff.
S2. A group of people are outside.

Fig. 1. An example of generating textual entailment.

We found three limitations in previous studies on GTE (which are analyzed detailedly in the related work section):

1. Rule-based methods often lack adequate coverage. Moreover, the process of formulating inference rules is inefficient and requires special knowledge.

2. Current sequence-to-sequence models are prone to produce simple and short hypotheses which are fragile when faced with more complex premises, although they have gotten rid of dependence on hand-crafted rules.

3. All previous works on GTE were evaluated by an inappropriate metric, i.e. BLEU, or non-objective human annotators which make horizontal comparison among GTE models inconvenient.

To circumvent these limitations, we firstly improve the sequence-to-sequence model by using residual LSTM which is a more potent encoder, leading to a state-of-art result with an improvement of correct rate by 3 percent over strong baseline models; and secondly propose a new assessment metric called EBR, which could evaluate different GTE models in an objective and fair way without human effort while also considering the diversity of generated hypotheses.

Moreover, we point out the limitation of the current sequence-to-sequence framework in GTE tasks, with some proposals for future research, hoping to generate more public discussion.

In the rest of this paper, we will describe the details of our improved GTE model in section 2; propose the EBR metric which is then compared with previous evaluation criteria on GTE, i.e. BLEU and human annotation, in section 3; introduce our experiments settings and results analyses in section 4; draw conclusions and discuss future research directions in the last section.

2 The Improved Sequence-to-Sequence Model for GTE

2.1 A Generic Encoder-Decoder Framework

The encoder-decoder framework was proposed by Cho et al. [3] for sequence-to-sequence NLP tasks, such as machine translation and dialogue generation. Similarly, it is intuitive to employ this model to generate textual entailment.

Fig. 2 shows a generic encoder-decoder framework for GTE. Each box in the illustration represents a cell of LSTM, around which arrows with a solid line indicate its inputs and outputs. The first three cells that share weights constitute the encoder whose duty is to “remember” the semantic information of the premise, while the remaining forms the decoder which is responsible for inferring a hypothesis. At each time step, the encoder takes a word in the premise as input and pass the cell states to the next step.

After receiving the encoding represented by a dense vector, the decoder starts to infer the first token of hypothesis and then takes the word just generated as input to produce more tokens until reaching a <EOS>.

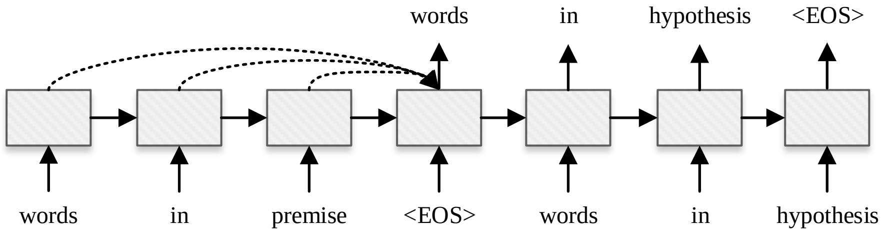


Fig. 2. Encoder-decoder framework for generating textual entailment.

2.2 Problems in Current Models

The above describes the basic architecture for generating inference from a single text. However, the memory of a fixed length vector is limited. Kolesnyk et al. [7] improved it by adding the word-by-word attention (the arrows with a dotted line[1]) which let the decoder reference the outputs of the encoder at each time step when generating tokens. By this mean, the decoder receives more information from the premise, which improved the correctness of inference.

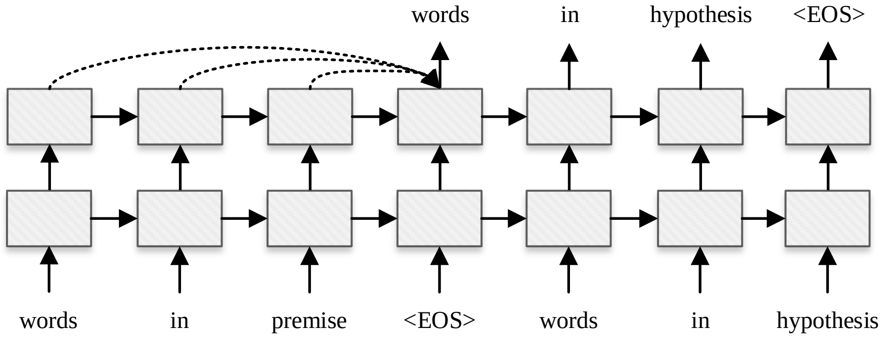


Fig. 3. 2-layer LSTM GTE Model [7].

We reimplemented the model proposed by Kolesnyk as our baseline, which is a 2-layer unidirectional LSTM sequence-to-sequence model, as depicted in **Fig. 3**. After reviewing the produced premise-hypothesis pairs, we found that the baseline model is

[1] For clarity, only one step of attention is drawn in figures.

prone to generate invalid sentences, e.g. S4 in **Fig. 4**, when faced with complex enough premises, e.g. S3, although it performs well on simple input sentences.

S3. A female gymnast in black and red being coached on bar skill.
S4. A female is in a bar.

Fig. 4. Examples of invalid sentence pairs generated by the baseline model.[2]

2.3 Our Improved GTE model by Residual LSTMs

We conjecture that the problems may lie in the unpowerful encoder which fails to encode some essential information and finally cause an invalid generating. In fact, when stacking multiple layers of neurons, such as LSTMs, the network often suffers from a degradation problem [5]. Residual connections are proved to be helpful to overcome this issue in an encoder-decoder framework [12]. We suspect that the degradation problem might be the major factor causing the invalid generating and add residual connections (arrows with a dashed line) to our sequence-to-sequence model, as shown in **Fig. 5** and **Fig. 6**(b). We will show by experiments that this modification is effective to alleviate the degradation problem, leading to a more informative generating with a much higher correct rate.

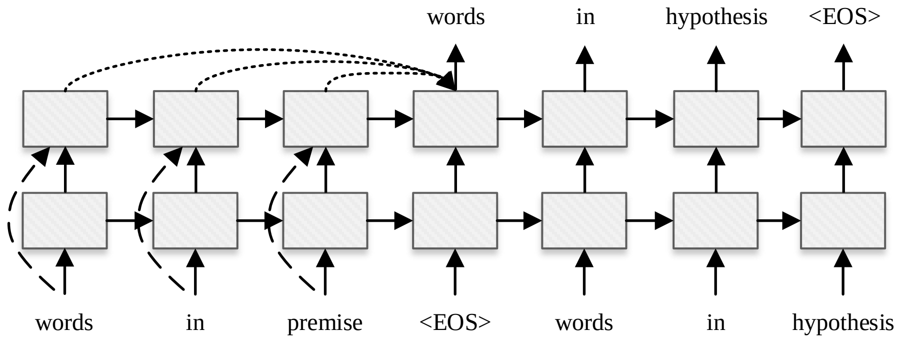


Fig. 5. 2-layer Residual LSTM GTE Model (Our model).

We improve the original 2-layered LSTM network (see **Fig. 6**(a)) by adding a residue shortcut connection. In our case, the residual connection performs an identity mapping $I(\cdot)$, followed by a pointwise addition (see **Fig. 6**(b), Eq. (2)).

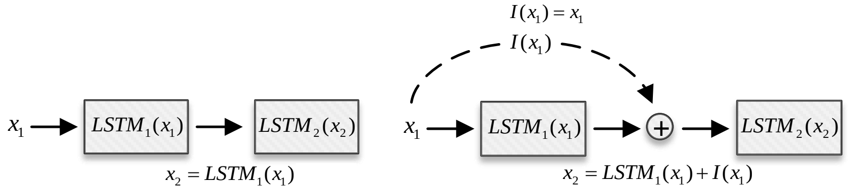
Formally, the output of a 2-layered Residual LSTM at timestep t is

$$output^t_{ResLSTM} = LSTM^t_2(x^t_2) \tag{1}$$

[2] Sentence pair S3-S4 is extracted from Kolesnyk's paper.

$$x_2^t = LSTM_1^t(x_1^t) + I(x_1^t) \tag{2}$$

where $I(x_1^t) = x_1^t$.



(a) 2-layer vanilla LSTM.

(b) 2-layer LSTM with residual connection.

Fig. 6. Residual connection for LSTMs at one timestep.

3 An Objective and Fair Metric for GTE: EBR

All the previous works on GTE use either human judgment or BLEU to evaluate the generated hypotheses by their models.

BLEU is designed as a metric to evaluate machine translation [10]. It considers exact matching between system generated translations and reference translations by counting n-gram overlaps. However, compared with machine translation, generated hypotheses have a greater diversity in both sentential form view and semantic view. As shown in **Fig. 7**, sentence S6 and S7 are both valid hypotheses inferred from the premise S5. However, there is no overlap n-gram between them, which leads to a zero-valued BLEU score. This phenomenon makes the use of BLEU in GTE settings problematic.

On the other hand, annotation by a human is inefficient when the test set contains thousands of examples, and random sampling may lead to instability of the evaluation results.

S5. Two young children in blue jerseys, one with the number 9 and one with the number 2 are standing on wooden steps in a bathroom and washing their hands in a sink.
S6. Two kids wearing numbered jerseys wash their hands.
S7. The children are in a bathroom.

Fig. 7. Examples of various valid hypotheses inferred from a single text.

To overcome these limitations, we propose our novel evaluation metric EBR, which is an abbreviation of Evaluation By RTE. As the name implies, we employ recognizing textual entailment systems to evaluate whether the generated hypotheses could be inferred from given premises.

Compared with BLEU, the diversity of hypotheses is considered instinctively by the design of RTE systems. Most RTE systems could adapt to the variety of hypotheses inferred from a single premise in both sentential form view and semantic view.

Thanks to modern hardware and optimized algorithms, EBR could validate GTE results more efficiently than human judgment. However, a possible shortage in EBR is that the accuracy of an RTE system is often lower than human annotator. We admit that every RTE system has its blind spot, e.g., knowledge-based RTE system may get stuck due to a lack of inference rules, while classifier-based RTE system may suffer from a different distribution over the test set with the training data. Nevertheless, the shallow left behind by an RTE system might be illuminated by another one. If we employ a bunch of RTE systems which are designed from different perspectives, they might light up the whole area just like a surgical lighting system. Inspired by this idea, EBR is designed to use an ensemble of existing RTE systems to measure the correctness of generated hypotheses.

The choice of RTE systems is essential to evaluate the produced hypotheses. After a survey of public available RTE systems, we choose the Excitement Open Platform (EOP) as our testbed [8]. The EOP is an open source state-of-the-art RTE platform including several heterogeneous RTE algorithms: transformation-based (BIUTEE), edit-distance based (EDIT), and classifier-based (TIE). In addition to these out-of-the-box EDAs, we use an ensemble version of them by majority voting (MAJOR). Finally, our EBR metric consists of a tuple of scores: {BIUTEE, EDIT, TIE, MAJOR}. Also, it is easy to extend EBR by employing more RTE techniques.

Another benefit brought by EBR is the independence on reference hypotheses, which make it possible to evaluate GTE results on unlabeled data.

4 Experiments and Analyses

4.1 Dataset

Our dataset is extracted from the Stanford Natural Language Inference (SNLI) corpus [1], which contains about 560K sentence pairs labeled as entailment, contradiction and neutral. We only use the entailment-labeled part, which contains 183,416 premise-hypothesis pairs in the training set, 3,329 pairs in the validation set, and 3,368 pairs in the test set.

4.2 Experiment Settings

We adapt glove 840B 300d vectors [11] as our word representation, with out-of-vocabulary words randomly initialized by sampling values uniformly from $[-\sqrt{3}, \sqrt{3}]$. Dropout (=0.5) is applied after each LSTM layer. Greedy decoding is used at test time. The dimension of LSTM units is fixed to 300 across all models, which are trained by the SGD algorithm (learning rate 0.3) with the mini-batch size of 64. All RTE systems in the EBR metric are executed by loading the pre-trained model published in their official website without retraining or fine-tuning.

4.3 Baseline Models

We implemented the GTE model proposed by Kolesnyk as our first baseline, which is a 2-layer unidirectional LSTM sequence-to-sequence model, as shown in **Fig. 3**. Furthermore, Bidirectional LSTM encoder (as depicted in **Fig. 8**) which is popular in RTE tasks [2] is also implemented as another baseline.

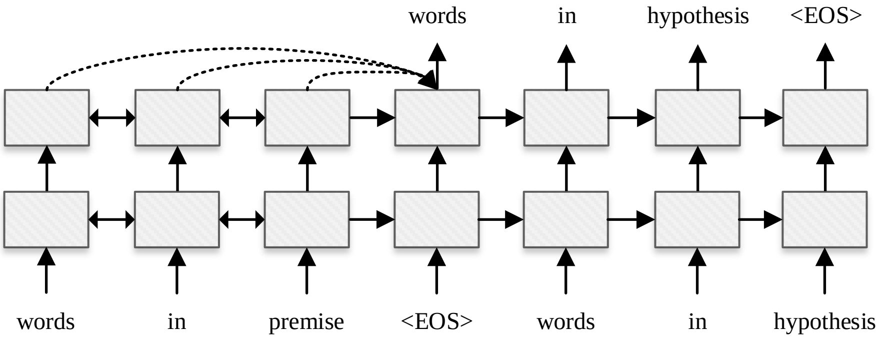


Fig. 8. 2-layer BiLSTM GTE Model (Inspired by Chen et al. [2]).

4.4 Results and Analyses

Table 1. Correct rate (%) of GTE models.

Models	EBR				Manual
	BIUTEE	EDIT	TIE	MAJOR	
Nevěřilová	-	-	-	-	47.06
Kolesnyk	65.41	73.21	69.83	71.61	80.00
BiLSTM	66.26	74.34	72.13	74.02	81.00
ResLSTM	**67.39**	**78.24**	**75.31**	**77.14**	**84.00**

Table 1 shows the GTE results evaluated by EBR and human judgment[3], where Nevěřilová[4] represents a rule-based GTE approach [9]; Kolesnyk (2-layer unidirectional LSTM, **Fig. 3**) and BiLSTM (see **Fig. 8**) are our baseline models; ResLSTM is our improved sequence-to-sequence model. It is ideal to employ annotators to assess all the generated sentences, but due to the huge volume of our test dataset, human assessment is done by random sampling 300 produced hypotheses (100 for each approach).

The results demonstrate that our improved model outperforms other baseline models on both EBR and Manual metrics. Another conclusion drawn from the table is that EBR

[3] BELU, which is proved to be inappropriate in the context of GTE (See Section 3), is excluded.
[4] This model is not reimplemented. The number 47.06 is obtained from the original paper.

metric behaves in a rather consistent way with a human annotator, where they all believe ResLSTM performs better than BiLSTM and then a unidirectional one. This phenomenon suggests that EBR is a more reasonable replacement for inefficient human annotators.

After reviewing the hypotheses produced by all approaches above, we found that hypotheses generated by our residual model (e.g. the last column in **Fig. 9**) usually keep more accurate information from the same premises (e.g. the first column in **Fig. 9**) than baselines (e.g. the middle columns in **Fig. 9**), which confirms our conjecture of bottleneck lying in the unpowerful encoder. In addition, the residual connections which keep accurate information from premises make our model more robust when faced with some complex sentences (e.g. the last two rows in **Fig. 9**).

Premise	Kolesnyk	BiLSTM	ResLSTM
The girl wearing a brown jacket whilst walking in snow.	The girl is walking. *(correct, **less informative**)*	The girl is outside. *(correct, **less informative**)*	A girl is walking outside. *(correct, more informative)*
Three young women perform a dance in a crowded hall.	Three women dance. *(correct, **less informative**)*	Three women are dancing. *(correct, **less informative**)*	Three women dance in a crowded room. *(correct, more informative)*
Two middle-aged police officers watch over a parking lot, at night.	Two police are watching a race. ***(incorrect)***	The police officers are at night. *(correct, **less informative**)*	Two police officers are looking at a parking lot. *(correct, more informative)*
A mass of people looking and moving in the same general direction.	People are in the distance. ***(incorrect)***	The people are in the area. ***(incorrect)***	A group of people are in the same direction. *(correct)*

Fig. 9. Examples of generated hypotheses from the same premise.

5 Related Work

To the best of our knowledge, Jia [6] is the first researcher to study how to produce an entailed sentence from a given premise. He proposed a naïve rule-based algorithm repeating pattern matching and applying sentence rewriting rules developed by experts. Although these rules are reliable, the process of formulating entailment rules requires special knowledge and is inefficient. According to the author, only ten rules could be made up manually by an expert in one hour, which make it unacceptable in practical applications. Nevěřilová [9] developed a similar rule-based method which also suffers from the lack of inference knowledge, with the result that only 47.06% of generated hypotheses are correct.

Sequence-to-sequence recurrent neural networks were first proposed for machine translation by Cho et al. [3]. They use an LSTM network to encode the sentence in source language as a fixed-length vector, and then another LSTM network to decode the target translation from it. Their method is an end-to-end approach so that no rules are involved, which exactly meet our needs to eschew the lack of inference rules. Kolesynk et al. [7] adapted the sequence-to-sequence framework from Cho for the GTE task. Furthermore, they employed a word-by-word attention, which allows the decoder to search in the encoder's outputs to avoid the memory bottleneck of the vanilla LSTM networks. Their model gets rid of the dependence on human-crafted inference rules, which makes it possible in practical applications. However, after reviewing their generated sentences, we found that the hypotheses produced by their model are often short and fragile when faced with more complex premises. We suspect that the problem may lie in the unpowerful sentence encoder, and our experiments confirm our assumption.

Prakash et al. [12] first proposed the stacked residual LSTM networks as sentence encoder by adding residual connections between vertically stacked multi-layer LSTM networks where the output of the previous layer of LSTM is fed to the input of the next one. Toderici et al. [13] used residual GRU to show an improvement in image compression rates for a given quality over JPEG. Our improved model is highly inspired by these RNN networks with residual connections.

All the works above on GTE use either human judgment or BLEU to evaluate the generated hypotheses by their models.

6 Conclusion and Future Work

In this paper, we described an improved sequence-to-sequence model with stacked residual LSTM networks and a novel evaluation metric EBR for the task of generating textual entailment. Experiments show that our improved model obtains state-of-the-art results and the EBR metric could validate various GTE models' performance efficiently in an objective and fair way.

We notice that there are also limitations in current sequence-to-sequence GTE models:

Firstly, hypotheses produced by current models are short for variety. Compared with the large space of possible valid hypotheses (see **Fig. 9**), only one sentence could be decoded by current sequence-to-sequence architecture, which is undesirable. Thus, how to increase the diversity of generation is a topic worthy of study.

Secondly, the hypotheses are generated blindly. In other words, the generation is performed without a purpose. For example, considering sentence S5 in **Fig. 9**, if the question is "what are the children doing?", sentence S6 should be generated. However, if the question is "where are the kids?", then a generation of S7 is more acceptable. Therefore, how to produce a hypothesis to meet some predefined requirements is another subject worthwhile to research.

As for the EBR metric, there is an inherent limitation – the correctness is upper-bounded by the correctness of the underlying classifiers. Although this problem is partially alleviated by an ensemble of heterogeneous RTE systems, a recognizing technique of reasoning relations with higher accuracy is always desirable.

We plan to study further along these directions.

Acknowledgements. This paper was supported by the National Natural Science Foundation of China (Grant No. 61472105, 61472107), The National High Technology Research and Development Program of China (863 Program) (2015AA015407).

References

1. Bowman, S.R. et al.: A large annotated corpus for learning natural language inference. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2015). doi:10.18653/v1/d15-1075
2. Chen, Q. et al.: Enhancing and Combining Sequential and Tree LSTM for Natural Language Inference. ArXiv160906038 Cs. (2017).
3. Cho, K. et al.: Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. ArXiv14061078 Cs Stat. (2014). doi:10.3115/v1/d14-1179
4. Dagan, I. et al.: The PASCAL Recognising Textual Entailment Challenge. Presented at the (2006). doi:10.1007/11736790_9
5. He, K. et al.: Deep Residual Learning for Image Recognition. 2016 IEEE Conf. Comput. Vis. Pattern Recognit. CVPR. (2016). doi:10.1109/cvpr.2016.90
6. Jia, J.: The generation of textual entailment with NLML in an intelligent dialogue system for language learning CSIEC. In: Natural Language Processing and Knowledge Engineering, 2008. NLP-KE'08. International Conference on. pp. 1–8 IEEE (2008). doi:10.1109/nlpke.2008.4906806
7. Kolesnyk, V. et al.: Generating Natural Language Inference Chains. ArXiv Prepr. ArXiv160601404. (2016).
8. Magnini, B. et al.: The Excitement Open Platform for Textual Inferences. In: ACL (System Demonstrations). pp. 43–48 (2014). doi: 10.3115/v1/p14-5008
9. Nevěřilová, Z.: Paraphrase and Textual Entailment Generation. In: International Conference on Text, Speech, and Dialogue. pp. 293–300 Springer (2014). doi:10.1007/978-3-319-10816-2_36
10. Papineni, K. et al.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 311–318 Association for Computational Linguistics (2002). doi:10.3115/1073083.1073135
11. Pennington, J. et al.: Glove: Global Vectors for Word Representation. In: EMNLP. pp. 1532–43 (2014). doi:10.3115/v1/d14-1162
12. Prakash, A. et al.: Neural Paraphrase Generation with Stacked Residual LSTM Networks. ArXiv161003098 Cs. (2016).
13. Toderici, G. et al.: Full Resolution Image Compression with Recurrent Neural Networks. ArXiv Prepr. ArXiv160805148. (2016).