SEQUENTIAL MATCHING MODEL FOR END-TO-END MULTI-TURN RESPONSE SELECTION

Qian Chen, Wen Wang

Speech Lab, DAMO Academy, Alibaba Group
{tanqing.cq, w.wang}@alibaba-inc.com

ABSTRACT

Multi-turn conversation understanding is an important challenge for building intelligent dialogue systems, and end-to-end multi-turn response selection is one of the major tasks. Previous state-of-the-art models used hierarchy-based (utterance-level and token-level) neural networks to explicitly model the interactions among the different turns' utterances for context modeling. In this paper, we demonstrate that the potentials of sequential matching approaches have not yet been fully exploited in the past for multi-turn response selection. We investigate a sequential matching model based only on chain sequence for multi-turn response selection. The proposed model outperforms all previous models, including previous state-of-the-art hierarchy-based models, and achieves new state-of-the-art performances on two large-scale public multi-turn response selection benchmark datasets.

Index Terms— multi-turn response selection, end-to-end, ESIM, neural network

1. INTRODUCTION

Dialogue systems are gaining more and more attention due to their encouraging potentials and commercial values. With the recent success of deep learning models [1, 2], building an end-to-end dialogue system became feasible. However, building end-to-end multi-turn dialogue systems is still quite challenging, requiring the system to remember and comprehend multi-turn conversation context, rather than only considering the current utterance as in single-turn dialogue systems.

Multi-turn dialogue modeling can be divided into generation-based methods [3, 4] and retrieval-based methods [5, 6]. The latter is the focus of this paper. Retrieval-based methods select the best response from a candidate pool for multi-turn context, which can be considered as performing a multi-turn response selection task. The typical approaches for multi-turn response selection mainly consist of sequence-based methods [5, 7] and hierarchy-based methods [8, 6, 9, 10]. Sequence-based methods usually concatenate the context utterances into a long sequence. Hierarchy-based methods normally model each utterance individually and then explicitly model the interactions among the utterances.

Recently, previous work [6, 9] claims that hierarchy-based methods with more complicated networks can achieve significant gains over sequence-based methods. However, in this paper, we investigate the efficacy of a sequence-based method, i.e., the Sequential Inference Model (ESIM) [11] originally developed for the natural language inference (NLI) task. The proposed approach outperforms all previous models, including previous state-of-the-art hierarchy-based methods, on two benchmark datasets, the Ubuntu [5] and E-commerce datasets [9], suggesting that the potentials of such sequential matching approaches have not been fully exploited in the past.

Hierarchy-based methods usually use extra neural networks to model the multi-turn utterances' relationship explicitly. They also usually need to truncate the utterances in the multi-turn context to make them the same length and shorter than the maximum length. However, the lengths of different turns usually vary significantly in real tasks. When using a large maximum length, we need to add a lot of zero padding in hierarchy-based methods, which will increase computation complexity and memory cost drastically. When using a small maximum length, we may throw away some important information in the multi-turn context. The contributions of this paper lie in two aspects. First, we propose to use a sequence-based model, the ESIM model, in the multi-turn response selection task to effectively address the above mentioned problem encountered by hierarchy-based methods. We concatenate the multi-turn context as a long sequence, and convert the multi-turn response selection task into a sentence pair binary classification task, i.e., whether the next sentence is the response for the current context. Compared to hierarchy-based methods, the ESIM model does not use extra neural networks to model the multi-turn utterances' relationship. Instead, the relationship is modeled by ESIM implicitly. Also, the ESIM model has less padding than hierarchy-based methods and hence has lower computational complexity and memory cost, since it does not require each utterance to have the same length. For the second contribution, we investigate different hyperparameters to make ESIM suitable for the multi-turn response selection task since the multi-turn response selection task is quite different from NLI tasks. NLI tasks need to determine whether a premise sentence can infer a hypothesis sentence. Compared to the context length in the standard NLI applications, the multi-turn response sentence task commonly has a much longer context (up to 500 words or more), due to concatenation of the multi-turn context. In our work we find the effective hyperparameters for the ESIM model for multi-turn response selection, such as applying word embedding pre-trained from training set and truncating the context in a reverse direction.

2. RELATED WORK

The previous work of modeling multi-turn response selection can be categorized into three categories, i.e., sentence-encoding based models, sequence-based matching models, and hierarchy-based models.

Sentence-encoding based models use Siamese architecture [12]. Lowe et al. [5] concatenated all utterances as the context representation and then computed the matching degree score based on sentence encoding methods, such as TF-IDF, RNN and LSTM. Kadlec et al. [13] used similar frameworks with CNN and BiLSTM.

Sequence-based matching models usually use attention mech-

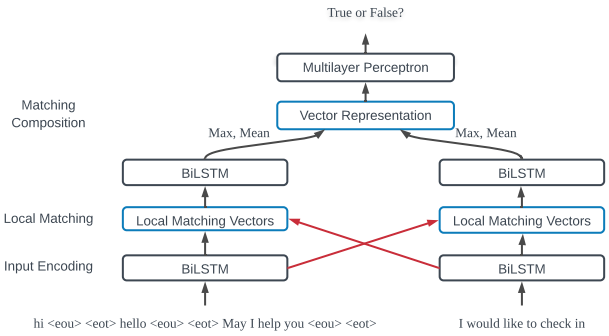

Fig. 1. A high-level flow diagram of the ESIM model.

anism to compare the token-level relationship between the context and the response, such as MV-LSTM [14], Matching-LSTM [15], Attentive-LSTM [16], Multi-Channels [6].

Hierarchy-based models often employ more complicated networks to model the token-level and utterance-level information explicitly. Zhou et al. [8] performed context-response matching with a multi-view model on both word level and utterance level. Wu et al. [6] used CNN to integrate the utterance-response matching information. Zhang et al. [9] proposed a deep utterance aggregation model to form a fine-grained context representation based on self-matching attention. Wu et al. [10] investigated matching a response with its multi-turn context using dependency information based entirely on attention by the Transformer structure [17].

Although the state-of-the-art systems [6, 9, 10] claim that the hierarchy-based methods outperform the sequence-based methods, they didn't compare with one of the state-of-the-art models developed for natural language inference, i.e., ESIM [11]. ESIM is proved to be powerful on the Stanford Natural Language Inference dataset [18], which needs to determine whether a hypothesis sentence can be inferred from a premise sentence.

3. MODEL DESCRIPTION

The multi-turn response selection task is to select the next utterance from a candidate pool, given a multi-turn context. We solve the problem by converting it to a binary classification task, which is similar to previous work [5, 6]. Given a multi-turn context and a candidate response, our model needs to determine whether the candidate response is the proper next utterance. In this section, we will introduce our model, which is originally developed for natural language inference, i.e., Enhanced Sequential Inference Model (ESIM) [11]. The model consists of three main components, i.e., input encoding, local matching, and matching composition, as shown in Figure 1.

3.1. Input Encoding

Instead of encoding the context information through more complicated hierarchical structures, the context information is simply encoded as follows. The multi-turn context is concatenated as a long sequence, which is denoted as $\boldsymbol{c} = (c_1, \ldots, c_m)$. The candidate response is denoted as $\boldsymbol{r} = (r_1, \ldots, r_n)$. Then we use the pre-trained word embedding $\mathbf{E} \in \mathbb{R}^{d_e \times |V|}$ to convert $\boldsymbol{c}$ and $\boldsymbol{r}$ to two vector sequences $[\mathbf{E}(c_1), \ldots, \mathbf{E}(c_m)]$ and $[\mathbf{E}(r_1), \ldots, \mathbf{E}(r_n)]$, where $|V|$ is the vocabulary size and d_e is dimension of the word embedding. To represent tokens in its contextual meaning, the context and the response are fed into BiLSTM encoders [19] to obtain context-dependent hidden states $\boldsymbol{c}^s$ and $\boldsymbol{r}^s$:

$$\boldsymbol{c}_i^s = \text{BiLSTM}_1(\mathbf{E}(\boldsymbol{c}), i)\,, \tag{1}$$

$$\boldsymbol{r}_j^s = \text{BiLSTM}_1(\mathbf{E}(\boldsymbol{r}), j)\,, \tag{2}$$

where i and j indicate the i-th token in the context and the j-th token in the response, respectively.

3.2. Local Matching

Modeling the local semantic relation between a context and a response is the critical component for determining whether the response is the proper next utterance. For instance, a proper response usually replies according to some keywords in the context, which can be obtained by modeling the local semantic relation. Instead of directly encoding the context and the response as two dense vectors, we use the attention mechanism to align the tokens from the context and response, and then calculate the semantic relation at the token level. The attention weight is calculated as:

$$e_{ij} = (\boldsymbol{c}_i^s)^{\mathrm{T}} \boldsymbol{r}_j^s\,. \tag{3}$$

Soft alignment is used to obtain the local relevance between the context and the response, which is calculated by the attention matrix $\mathbf{e} \in \mathbb{R}^{m \times n}$ in Equation (3). Then for the hidden state of the i-th token in the context, i.e., $\boldsymbol{c}_i^s$ (already encoding the token itself and its contextual meaning), the relevant semantics in the candidate response is identified as a vector $\boldsymbol{c}_i^d$ using e_{ij} by weighted combination of all the response's states, called dual vector here, more specifically as shown in Equation (4).

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{n} \exp(e_{ik})}\,,\ \boldsymbol{c}_i^d = \sum_{j=1}^{n} \alpha_{ij} \boldsymbol{r}_j^s\,, \tag{4}$$

$$\beta_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{m} \exp(e_{kj})}\,,\ \boldsymbol{r}_j^d = \sum_{i=1}^{m} \beta_{ij} \boldsymbol{c}_i^s\,, \tag{5}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{\beta} \in \mathbb{R}^{m \times n}$ are the normalized attention weight matrices with respect to the 2-axis and 1-axis. The similar calculation is performed for each token in the response, i.e., $\boldsymbol{r}_j^s$, with Equation (5) to obtain the dual vector $\boldsymbol{r}_j^d$.

By comparing vector pair $< \boldsymbol{c}_i^s, \boldsymbol{c}_i^d >$, we can model the token-level semantic relation between aligned token pairs. The similar calculation is also applied for vector pair $< \boldsymbol{r}_j^s, \boldsymbol{r}_j^d >$. We collect local matching information as follows:

$$\boldsymbol{c}_i^m = F([\boldsymbol{c}_i^s; \boldsymbol{c}_i^d; \boldsymbol{c}_i^s - \boldsymbol{c}_i^d; \boldsymbol{c}_i^s \odot \boldsymbol{c}_i^d])\,, \tag{6}$$

$$\boldsymbol{r}_j^m = F([\boldsymbol{r}_j^s, \boldsymbol{r}_j^d; \boldsymbol{r}_j^s - \boldsymbol{r}_j^d; \boldsymbol{r}_j^s \odot \boldsymbol{r}_j^d])\,, \tag{7}$$

where a heuristic matching approach [20, 11] with difference and element-wise product is used here to obtain local matching vectors $\boldsymbol{c}_i^m$ and $\boldsymbol{r}_j^m$ for the context and response, respectively. F is a 1-layer feed-forward neural network with the ReLU to reduce dimension.

3.3. Matching Composition

Matching composition is realized as follows. To determine whether the response is the next utterance for the current context, we need to

explore a composition layer to compose the local matching vectors ($\boldsymbol{c}^m$ and $\boldsymbol{r}^m$) collected above:

$$\boldsymbol{c}_i^v = \mathrm{BiLSTM}_2(\boldsymbol{c}^m, i)\,, \tag{8}$$

$$\boldsymbol{r}_j^v = \mathrm{BiLSTM}_2(\boldsymbol{r}^m, j)\,. \tag{9}$$

We also use BiLSTMs as building blocks for the composition layer, but the role of BiLSTMs here is completely different from that in the input encoding layer. The BiLSTMs here read local matching vectors ($\boldsymbol{c}^m$ and $\boldsymbol{r}^m$) and learn to discriminate critical local matching vectors for the overall utterance-level relationship.

Our model converts the output hidden vectors of BiLSTMs to the fixed-length vectors with pooling operations and feeds it to the final classifier to determine the overall relationship. Particularly, we use max and mean pooling and concatenate them all to get a fixed-length vector. Then the final vector are fed to the multi-layer perceptron (MLP) classifier.

$$y = \mathrm{MLP}([\boldsymbol{c}_{max}^v; \boldsymbol{c}_{mean}^v, \boldsymbol{r}_{max}^v; \boldsymbol{r}_{mean}^v])\,. \tag{10}$$

The MLP has one hidden layer with *tanh* activation and *softmax* output layer. The entire model is trained via minimizing the cross-entropy loss in an end-to-end manner.

4. EXPERIMENTS

4.1. Datasets

We evaluate our model on two large-scale muli-turn response selection benchmarks, i.e., the Ubuntu dataset [5] and E-commerce dataset [9]. Data statistics are summarized in Table 1.

4.1.1. Ubuntu dataset

The Ubuntu dataset consists of multi-turn conversations constructed from Ubuntu Internet Relay Chat (IRC) logs. The training set contains 1 million context-response pairs and the ratio between positive responses and negative responses is 1:1. On both development and test sets, each context is associated with one positive response and 9 negative responses. Recall at position k (R@k) is selected as the metrics, i.e., R@1, R@2 and R@5, remaining the same as in the previous work [5, 9, 10].

4.1.2. E-commerce dataset

The E-commerce dataset [9] is collected from real-word conversations between customers and customer service staff from Taobao[1], which is the largest e-commerce platform in China. The negative responses are selected by ranking the response corpus based on the last utterance along with the top-5 keywords in the context using Apache Lucene[2]. The ratio between positive responses and negative responses is 1:1 in both training and development sets, and 1:9 in the test set. R@1, R@2 and R@5 are also selected as the metrics which are the same as in Zhang et al. [9].

4.2. Training Details

The multi-turn context is concatenated and two special tokens __eou__ and __eot__ are inserted, where __eou__ indicates end-of-utterance and __eot__ indicates end-of-turn. The two datasets were already tokenized when released, and we did not apply any further pre-processing. We use pre-trained word embedding on the training data by the word2vec tool [21]. Adam [22] is used for optimization with an initial learning rate of 0.0002 for the Ubuntu dataset, and 0.0004 for the E-commerce dataset. The batch size is 16 for the Ubuntu dataset, and 32 for the E-commerce dataset. The hidden size of BiLSTMs and MLP is set to 300 for both datasets. To make the sequences shorter than the maximum length, we cut off last tokens for the response but did it in the reverse direction for the context, because the last few utterances in the context are more important than the first few utterances. For the Ubuntu dataset, we set the maximum length of the context to 400, and the maximum length of the response to 150. For E-commerce dataset, we set the maximum length of context to 300, and the maximum length of response to 50. The above hyperparameters are tuned based on the development set.

[1]https://www.taobao.com

[2]http://lucene.apache.org/

4.3. Results

The results on two benchmarks are summarized in Table 2. The first group of models includes sentence-encoding based methods. They use hand-craft features or neural network features to encode both context and response, then a cosine classifier or MLP classifier is applied to decide the relationship between the two sequences. Previous work used TF-IDF, RNN [5] and CNN, LSTM, BiLSTM [13] to encode the context and the response.

The second group of models consists of sequence-based matching models, which usually use the attention mechanism, including MV-LSTM [14], Matching-LSTM [15], Attentive-LSTM [16], and Multi-Channels [6]. These models compared the token-level relationship between the context and the response, rather than comparing the two dense vectors directly as in sentence-encoding based methods. These kinds of models achieved significantly better performance than the first group of models. However, the potentials of such sequential models have not been fully exploited in the past.

The third group of models includes more complicated hierarchy-based models, which usually model the token-level and utterance-level information explicitly. Multi-View [8] model utilized utterance relationship from the word sequence view and utterance sequence view. DL2R model [7] employed neural networks to reformulate the last utterance with other utterances in the context. SMN model [6] used CNN and attention to match a response with each utterance in the context. DUA [9] and DAM [10] applied a similar framework as SMN [6], where one improved with gated self attention and the other improved with the Transformer structure [17].

Although the previous work claimed that they achieved the state-of-the-art performance by using the hierarchical structure of multi-turn context, our proposed ESIM sequential matching model outperformed all previous results. On the Ubuntu dataset, the ESIM model achieved significant gains on performance, up to 79.6% (from 76.7%) R@1, 89.4% (from 87.4%) R@2 and 97.5% (from 96.9%) R@5. For the E-commerce dataset, the ESIM model also obtained substantial improvement, up to 57.0% (from 50.1%) R@1, 76.7% (from 70.0%) R@2 and 94.8% (from 92.1%) R@5. On one Tesla M40 GPU machine, the training time is 30 hours for 7 epochs on the Ubuntu dataset, and 14 hours for 12 epochs on the E-commerce dataset. These results demonstrate that the potentials of sequential matching models for multi-turn response selection have not been fully exploited yet in previous work.

4.4. Ablation Analysis

We further analyzed the major hyperparameters that are of importance to help us achieve this good performance. We trained a num-

Name	Ubuntu			E-commerce		
	Train	Dev	Test	Train	Dev	Test
# context-response pairs	1M	500K	500K	1M	10K	10K
# candidates per context	2	10	10	2	2	10
Average # tokens of concatenated context	135	134	135	49	49	51
Average # tokens of response	21	21	21	7	7	10
Vocabulary size	180K	180K	440K	36K	10K	6K

Table 1. Statistics of Ubuntu and E-commerce datasets.

Models	Ubuntu			E-commerce		
	R@1 (+%)	R@2 (+%)	R@5 (+%)	R@1 (+%)	R@2 (+%)	R@5 (+%)
TF-IDF [5]	0.410	0.545	0.708	0.159	0.256	0.477
RNN [5]	0.403	0.547	0.819	0.325	0.463	0.775
CNN [13]	0.549	0.684	0.896	0.328	0.515	0.792
LSTM [13]	0.638	0.784	0.949	0.365	0.536	0.828
BiLSTM [13]	0.630	0.780	0.944	0.355	0.525	0.825
MV-LSTM [14]	0.653	0.804	0.946	0.412	0.591	0.857
Match-LSTM [15]	0.653	0.799	0.944	0.410	0.590	0.858
Attentive-LSTM [16]	0.633	0.789	0.943	0.401	0.581	0.849
Multi-Channel [6]	0.656	0.809	0.942	0.422	0.609	0.871
Multi-View [8]	0.662	0.801	0.951	0.421	0.601	0.861
DL2R [7]	0.626	0.783	0.944	0.399	0.571	0.842
SMN [6]	0.726	0.847	0.961	0.453	0.654	0.886
DUA [9]	0.752	0.868	0.962	0.501	0.700	0.921
DAM [10]	0.767	0.874	0.969	-	-	-
Proposed ESIM	**0.796** (3.8)	**0.894** (2.3)	**0.975** (0.6)	**0.570** (13.8)	**0.767** (9.6)	**0.948** (2.9)

Table 2. Comparison of different models on two benchmark datasets. All the results except ours are cited from previous work [9, 10]. "+%" indicates the relative performance improvement over previous state-of-the-art results.

ber of ESIM models with different types of pre-trained word embedding, different maximum lengths of context and response, and truncating the context in the reverse direction or not, while for the rest using the same hyperparameters as described previously. Results on the Ubuntu development set are shown in Table 3. We can see that word2vec embedding trained on the training dataset achieves better results than Fasttext [23] embedding trained on the unlabeled corpus (Common Crawl) and random initialization, because the Ubuntu dataset has many rare in-domain words. In addition, the larger context or response length leads to strict performance improvement. This result demonstrates that ESIM can effectively utilize long multi-turn context information, which may be lost in hierarchy-based methods due to the limited utterance length. We can also see that truncating the context in the reverse direction leads to performance improvement, which shows that the last few utterances in context are more important than the first few utterances.

Hyperparams				Dev Result		
CtxLen	RepLen	Rev	Emb	R@1	R@2	R@5
400	150	Y	W2V	0.797	0.893	0.976
400	150	Y	Fasttext	0.776	0.876	0.970
400	150	Y	Random	0.732	0.844	0.958
300	150	Y	W2V	0.793	0.892	0.976
200	150	Y	W2V	0.793	0.891	0.976
100	150	Y	W2V	0.783	0.886	0.974
400	100	Y	W2V	0.795	0.893	0.976
400	50	Y	W2V	0.792	0.892	0.975
400	150	N	W2V	0.793	0.892	0.976
100	150	N	W2V	0.707	0.827	0.951

Table 3. Ablation over ESIM model on Ubuntu dataset. CtxLen = maximum length of context; RepLen = maximum length of response; Rev = truncate the context in reverse direction. Emb = the type of pre-trained word embedding.

5. CONCLUSION

Previous state-of-the-art multi-turn response selection models used hierarchy-based (utterance-level and token-level) neural networks to explicitly model the interactions among the different turns' utterances for context modeling. In this paper we demonstrate that a sequential matching model based only on chain sequence can outperform all previous models, including hierarchy-based methods, suggesting that the potentials of such sequential matching approaches have not been fully exploited in the past. Specially, we achieved new state-of-the-art performances on two large-scale public multi-turn response selection benchmarks, i.e., Ubuntu and E-commerce datasets. Future work on multi-turn response selection includes exploring the efficacy of external knowledge, such as knowledge graph and user profile. In addition, we will continue evaluating the effectiveness of the proposed approach on other large scale multi-turn response selection datasets.

6. REFERENCES

[1] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 3776–3784.

[2] Xiujun Li, Yun-Nung Chen, Lihong Li, Jianfeng Gao, and Asli Çelikyilmaz, "End-to-end task-completion neural dialogue systems," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017*, 2017, pp. 733–743.

[3] Iulian Vlad Serban, Tim Klinger, Gerald Tesauro, Kartik Talamadupula, Bowen Zhou, Yoshua Bengio, and Aaron C. Courville, "Multiresolution recurrent neural networks: An application to dialogue response generation," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 3288–3294.

[4] Ganbin Zhou, Ping Luo, Rongyu Cao, Fen Lin, Bo Chen, and Qing He, "Mechanism-aware neural machine for dialogue response generation," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 3400–3407.

[5] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau, "The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems," in *Proceedings of the SIGDIAL 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2015, pp. 285–294.

[6] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li, "Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, 2017, pp. 496–505.

[7] Rui Yan, Yiping Song, and Hua Wu, "Learning to respond with deep neural networks for retrieval-based human-computer conversation system," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016*, 2016, pp. 55–64.

[8] Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan, "Multi-view response selection for human-computer conversation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016*, 2016, pp. 372–381.

[9] Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu, "Modeling multi-turn conversation with deep utterance aggregation," in *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018*, 2018, pp. 3740–3752.

[10] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu, "Multi-turn response selection for chatbots with deep attention matching network," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018*, 2018, pp. 1118–1127.

[11] Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen, "Enhanced LSTM for natural language inference," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, 2017, pp. 1657–1668.

[12] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah, "Signature verification using a siamese time delay neural network," in *Advances in Neural Information Processing Systems 6*, 1993, pp. 737–744.

[13] Rudolf Kadlec, Martin Schmid, and Jan Kleindienst, "Improved deep learning baselines for ubuntu corpus dialogs," *CoRR*, vol. abs/1510.03753, 2015.

[14] Shengxian Wan, Yanyan Lan, Jun Xu, Jiafeng Guo, Liang Pang, and Xueqi Cheng, "Match-srnn: Modeling the recursive matching structure with spatial RNN," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016*, 2016, pp. 2922–2928.

[15] Shuohang Wang and Jing Jiang, "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*, 2016, pp. 1442–1451.

[16] Ming Tan, Bing Xiang, and Bowen Zhou, "Lstm-based deep learning models for non-factoid answer selection," *CoRR*, vol. abs/1511.04108, 2015.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Annual Conference on Neural Information Processing Systems 2017*, 2017, pp. 6000–6010.

[18] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning, "A large annotated corpus for learning natural language inference," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015*, 2015, pp. 632–642.

[19] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin, "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*, 2016.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean, "Distributed representations of words and phrases and their compositionality," in *27th Annual Conference on Neural Information Processing Systems 2013.*, 2013, pp. 3111–3119.

[22] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[23] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin, "Advances in pre-training distributed word representations," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018*, 2018.