DUAL PATH MODELING FOR SEMANTIC MATCHING BY PERCEIVING SUBTLE CONFLICTS

Chao Xue[♠,1], *Di Liang*[♣,1], *Sirui Wang*[♣] ,*Wei Wu*[♣],*Jing Zhang*[♠]

[♠] College of Software, Beihang University, Beijing, China
[♣]Centre for Natural Language Processing, Meituan Inc., Beijing, China

ABSTRACT

Transformer-based pre-trained models have achieved great improvements in semantic matching. However, existing models still suffer from insufficient ability to capture subtle differences. The modification, addition and deletion of words in sentence pairs may make it difficult for the model to predict their relationship. To alleviate this problem, we propose a novel Dual Path Modeling Framework to enhance the model's ability to perceive subtle differences in sentence pairs by separately modeling affinity and difference semantics. Based on dual-path modeling framework we design the Dual Path Modeling Network (**DPM-Net**) to recognize semantic relations. And we conduct extensive experiments on 10 well-studied semantic matching and robustness test datasets, and the experimental results show that our proposed method achieves consistent improvements over baselines.

Index Terms— dual path modeling, semantic matching, neural language processing, deep learning

1. INTRODUCTION

Semantic Sentence Matching (SSM) is a fundamental NLP task. It's goal is to compare two sentences and identify their semantic relationship. In paraphrase identification, SSM is used to determine whether two sentences are paraphrase or not [1]. In natural language inference task, SSM is utilized to judge whether a hypothesis sentence can be inferred from a premise sentence [2]. In the answer sentence selection task, SSM is employed to assess the relevance between query-answer pairs and rank all candidate answers [3].

Across the rich history of semantic sentence matching research, there have been two main streams of studies for solving this problem. One is to utilize a sentence encoder to convert sentences into low-dimensional vectors , and apply a parameterized function to learn the matching scores between them [3]. Another paradigm adopts attention mechanism to calculate scores between tokens from two sentences, and then the matching scores are aggregated to make a sentence-level decision [4, 5]. In recent years, pre-trained models, such as BERT [6], RoBERTa [7], have became much more popular and achieved outstanding performance in SSM. And they are shown to be powerful contextual representations for predicting sentence relations, as they capture richer linguistic hierarchies in sentences.

Although previous studies have provided some insights, existing models still suffer from insufficient ability to capture subtle differences. Figure 1 demonstrates a case suffers from this problem. Although the sentence pairs in this figure are semantically different, they are too similar in literal for those pre-trained language models to distinguish accurately. An important reason is that although the model can measure the matching degree in global semantics, it ignores the local subtle differences between texts. Because for text pairs with highly similar matching words, the overall semantic difference is often caused by different local differences. Furthermore, existing text matching models based on pretrained models are directly fine-tuned with the training data. It makes the model incapable of generalizing to text matching tasks with highly similar text formats, ultimately resulting in the model lacking the ability to capture fine-grained differences between new samples. Inspired by Sparsegen [8], we hypothesize that a more flexible model structure can help the model better understand the relationship of sentence pairs. In this paper, we focus on exploring the modeling of affinity and difference between texts to enhance the model's ability to understand fine-grained semantic differences. Therefore, two systemic questions arise naturally:

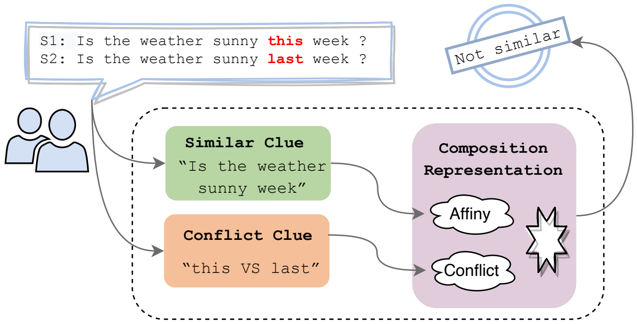

S1: Is the weather sunny this week ?
S2: Is the weather sunny last week ?
Not similar
Similar Clue
"Is the weather sunny week"
Conflict Clue
"this VS last"
Composition Representation
Affiny
Conflict

Fig. 1. The Dual Path Modeling Framework for semantic matching.

Q1: How to equip the model with the ability to model the affinity and difference between sentence pairs? We analyse that different kinds of attention are complementary clues for sentence matching, which can capture different levels of information in the text sequence. In this paper, we propose a dual attention module including a difference attention accompanied with the affinity attention. Affinity attention and difference attention aggregate word- and phrase-level interactions using dot-product cross-attention and subtraction-based cross-attention. And finally obtain semantic representations describing the affinity and difference, respectively.

Q2: How to fuse two types of semantic representations into a unified representation? We observe that simple aggregation with fixed or average importance weights may be detrimental to fusing heterogeneous vectors. We propose to adaptively aggregate the representations obtained by multiple attention functions from two perspectives. Firstly, the internal aggregation aggregates the matching information together with each word in the sentence in each attention function. Secondly, external aggregation combines the matching information of all attention functions. The output final vectors can better describe the matching details of sentence pairs.

[1] Equal contribution.

The main contributions of this work can be summarized as follows. First, we conduct an in-depth analysis of the subtle differences in semantic matching and propose a new dual-path modeling framework. Second, the proposed DPM-Net based DPM framework can effectively exploits and aggregates two complementary attention models, such that the intrinsic complex relationship between sentence pairs can be fully discovered for effective semantic matching. Finally, we conduct intensive experiments on 10 matching datasets and robustness testing datasets, and the results show that our method achieves consistent improvements across both architectures (representation-based and interaction-based).

2. RELATED WORK

2.1. Semantic Sentence Matching

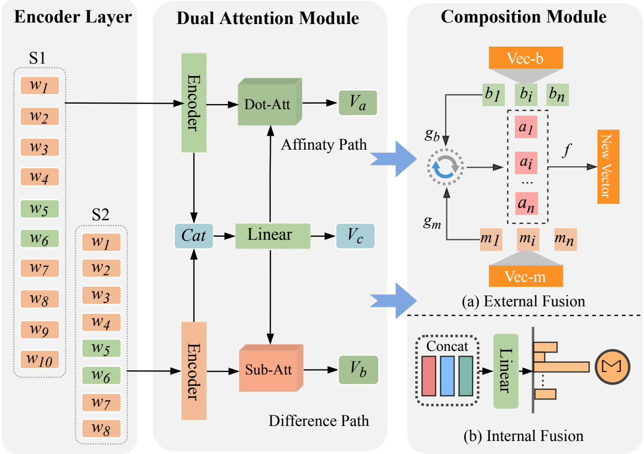
Semantic Sentence Matching is a fundamental task in NLP. In recent years, thanks to the appearance of large-scale annotated datasets [2], neural network models have made great progress in SSM [9], mainly fell into two categories. The first one [10, 11] focuses on encoding sentences into corresponding vector representations without any cross-interaction and applies a classifier layer to obtain similarity. The second one [12, 4] utilizes cross-features as an attention module to express the word-level or phrase-level alignments, and aggregates these integrated information to acquire similarity. Recently, the shift from neural network architecture engineering to large-scale pre-training has achieved outstanding performance in SSM and many other tasks. Meanwhile, leveraging external knowledge [13, 14] to enhance PLMs has been proven to be highly useful for multiple NLP tasks [15]. Therefore, recent work attempts to integrate external knowledge into pre-trained language models, such as AMAN, SemBERT, UERBERT, and so on [16, 17, 18, 19, 20, 21].

2.2. Robustness test

Although neural network models have achieved human-like or even superior results in multiple tasks, they still face the insufficient robustness problem in real application scenarios [22]. Tiny literal changes may cause misjudgments. Especially in some cases where fine-grained semantic needs to be discriminated. Besides, most of the current work utilizes one single metric to evaluate their model, may overestimate model capability and lack a fine-grained assessment of model robustness [22]. Therefore, recent work starts to focus on robustness research from multiple perspectives. TextFlint [22] incorporates multiple transformations to provide comprehensive robustness analysis. [23] provide an overall benchmark for current work on adversarial attacks. And [24] propose a more comprehensive evaluation system and add more detailed output analysis indicators.

3. TASK DEFINITION

Formally, we can represent each example of sentence pairs as a triple (Q, P, y), where Q = (q_1, ..., q_N) is a sentence with a length N, P = (p_1, ..., p_M) is another sentence with a length M, and y $\in$ Y is the label representing the relationship between Q and P. Specifically, for a paraphrase identification task, Q and P are two sentences, Y = 0, 1, where y = 1 means that Q and P are paraphrase of each other, and y = 0 otherwise. For a natural language inference task, Q is a premise sentence, P is a hypothesis sentence,and y=entailment, contradiction, neutral, where entailment indicates P can be inferred from Q, contradiction indicates P cannot be the true condition on Q, and neutral means P and Q are irrelevant to each other.



(a) External Fusion

(b) Internal Fusion

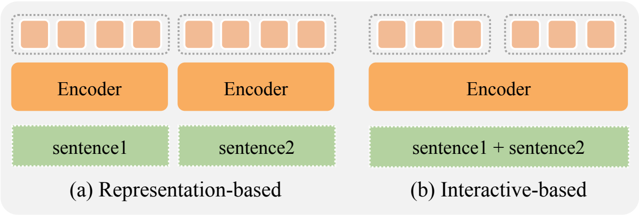
Fig. 2. The overall architecture of the DPM-Net.

4. METHOD

We show the design of the Dual Path Modeling Network in Figure 2. It consists of three parts under the dual path modeling framework. First, We use encoder(eg, transformer, Bert , Roberta) to obtain the context representation of two sentences through representation-based method or interaction-based method. Second, we use two different types of attention functions to model the interaction of sentence pairs from different perspectives. Next, we aggregate the matching information along with words in **P** an **Q** in two steps. We propose to adaptively aggregate representations obtained by dual attention functions from two perspectives. First, inner aggregation aggregates matching information with each word in the sentence in each attention function. Second, external aggregation combines the matching information of all attention features. We apply an aggregation mechanism to adaptively aggregate the two representations. Finally, we apply a Multilayer Perceptron (MLP) classifier for the final decision.

4.1. Encoder layer

For sentence pairs S1=$\{w_t^p\}_{t=1}^N$ and S2=$\{w_t^q\}_{t=1}^N$, we first convert the sentences into vector representations using an encoder. Since we want to explore the performance of DPM on representation-based and interaction-based methods, we use two methods to obtain text representations, respectively. The difference between the two methods is shown in Figure 3, taking Bert as an example. We then use a encoder to produce new representation Q=$\{q_1$,...,$q_n\}$ and P=$\{p_1$,...,$p_n\}$ of all words in two sentences respectively.



(a) Representation-based

(b) Interactive-based

At the same time, we concatenate the obtained two representations and perform linear transformation on them, and finally get V=$\{v_1$,...,$v_n\}$, Where N is the length after the text padding.

Table 1. Performance comparison of integrating DPM-Net in interaction-based methods on 10 Semantic Matching Benchmarks.

Model	Pre-train	MRPC	QQP	STS-B	MNLI-m/mm	QNLI	RTE	SNLI	Sci	SICK	Twi	Avg
Transformer†[25]	✗	81.7	84.4	70.4	72.3/71.4	80.3	58.1	81.7	70.6	-	-	-
Transformer+DPM(ours)†	✗	**81.9**	**85.1**	**71.8**	**72.7/72.4**	**80.9**	**59.4**	**85.2**	**77.3**	**-**	**-**	**-**
BERT-Base†[6]	✓	87.2	89.1	87.8	84.3/83.7	90.4	67.2	90.7	91.8	87.2	84.8	85.8
BERT-Base+DPM(ours)†	✓	**89.2**	**89.5**	**89.3**	**85.2/84.8**	**91.0**	**68.8**	**91.2**	**92.4**	**87.9**	**96.6**	**86.9**
BERT-Large†[6]	✓	88.9	89.3	86.6	86.8/86.3	92.7	70.1	91.0	94.4	91.1	91.5	88.0
BERT-Large+DPM(ours)†	✓	**89.6**	**89.7**	**88.3**	**86.9/86.7**	**93.2**	**72.5**	**91.4**	**94.5**	**91.4**	**92.0**	**88.7**
RoBERTa-Base†[7]	✓	89.3	89.6	87.4	86.3/86.2	92.2	73.6	90.8	92.3	87.9	85.9	87.6
RoBERTa-Base+DPM(ours)†	✓	**89.9**	**91.0**	**88.6**	**87.6/87.2**	**93.6**	**81.1**	**91.5**	**93.7**	**89.3**	**87.3**	**89.1**
RoBERTa-Large†[7]	✓	89.4	89.7	90.2	89.5/89.3	92.7	83.8	91.2	94.3	91.2	91.9	90.3
RoBERTa-Large+DPM(ours)†	✓	**90.2**	**91.3**	**90.8**	**90.2/90.1**	**94.0**	**84.2**	**91.8**	**94.8**	**90.8**	**92.3**	**90.9**

Table 2. Performance comparison of integrating DPM-Net in representation-based methods on 10 Semantic Matching Benchmarks.

Model	Pre-train	MRPC	QQP	STS-B	MNLI-m/mm	QNLI	RTE	SNLI	Sci	SICK	Twi	Avg
Transformer†[25]	✗	71.5	79.6	66.2	66.7/66.5	75.8	59.2	74.3	69.9	-	-	-
Transformer+DPM(ours)†	✗	**73.4**	**83.2**	**69.4**	**68.3/68.2**	**77.7**	**59.8**	**80.1**	**72.4**	**-**	**-**	**-**
BERT-Base†[6]	✓	81.4	82.6	81.3	78.8/78.4	84.4	60.2	83.3	89.8	80.9	79.1	80.4
BERT-Base+DPM(ours)†	✓	**83.6**	**84.4**	**85.1**	**79.6/79.4**	**86.0**	**63.6**	**86.1**	**90.9**	**82.5**	**82.7**	**82.1**
BERT-Large†[6]	✓	82.5	83.4	83.8	80.3/79.9	86.1	67.9	86.8	90.6	84.2	81.7	82.5
BERT-Large+DPM(ours)†	✓	**83.3**	**84.9**	**86.3**	**81.6/81.1**	**87.5**	**70.1**	**87.2**	**91.3**	**84.9**	**83.7**	**83.8**
RoBERTa-Base†[7]	✓	82.3	82.7	82.2	79.1/78.9	85.8	65.6	84.5	90.6	82.4	81.6	81.4
RoBERTa-Base+DPM(ours)†	✓	**83.9**	**85.2**	**85.9**	**79.5/79.3**	**87.1**	**67.7**	**86.8**	**91.5**	**82.9**	**82.8**	**82.9**
RoBERTa-Large†[7]	✓	83.4	83.8	84.2	81.3/81.1	86.5	68.9	87.6	91.2	84.8	82.6	83.2
RoBERTa-Large+DPM(ours)†	✓	**84.1**	**85.9**	**87.5**	**82.4/82.2**	**88.0**	**71.3**	**88.4**	**92.5**	**85.6**	**84.8**	**84.7**

4.2. Dual attention module

In dual attention module, we use two different attention functions to model the semantic relationship between sentence pairs from different perspectives. The input of the multi-view attention module is a triple of P, Q, $V \in R^{d_{seq}} \times d_v$, where d_v is the latent dimension, d_{seq} is the length of the utterance. We use p_i, q_i and v_i to denote the i-th tokens of P, Q, and V respectively.

4.2.1. Dot Attention

Dot attention is the most commonly used attention mechanism in semantic correlation modeling. And it follows the standard dot-product attention that the transformer operates by default. For the sake of simplicity, the formulations of it not be repeated here, please refer to [6] for more details. We denote the output vector as:

$$s_j^t = \mathbf{q}_j \odot \mathbf{v}_t, \quad a_i^t = \frac{exp(s_i^t)}{\sum_{j=1}^{N} exp(s_j^t)}, \quad \mathbf{q}_t^d = \sum_{i=1}^{N} a_i^t \mathbf{q}_i \qquad (1)$$

where $\mathbf{q}_t^d \in R^{1\times d_v}$ is the output of the t-th position obtained after the dot attention calculation and $\odot$ is element-wise dot product.

4.2.2. Subtract Attention

The second part of dual attention module is the subtract attention that captures and aggregates the difference information between sentence pairs. It allows the model to pay attention to dissimilar parts between sentence pairs by element-wise subtraction as:

$$s_j^t = \tanh(\mathbf{W}_m(\mathbf{p}_j - \mathbf{v}_t)), a_i^t = \frac{exp(s_i^t)}{\sum_{j=1}^{N} exp(s_j^t)}, \mathbf{q}_t^s = \sum_{i=1}^{N} a_i^t \mathbf{p}_i \quad (2)$$

where $\mathbf{q}_t^s \in R^{1\times d_v}$ is the output of the t-th position obtained after the minus attention calculation, and $\mathbf{W}_m \in R^{d_{seq}\times d_v}$ are trainable parameters.

4.3. Composition module

The composition module is divided into two stages, one is internal aggregation and the other is external aggregation.

4.3.1. Internal Aggregation

Internal aggregation is to integrate the representation obtained after attention with the original representation. For each position t, we concatenate v_t with the representation q_t^c obtained by attention, and then use gating to scale the overall information,c = (d,s). As shown below, This is an example of internal integration of dot attention:

$$\mathbf{x}_t^d = \left[q_t^d, v_t\right], \quad g_i = \sigma\left(\mathbf{W}_g \mathbf{x}_t^d\right) \qquad (3a)$$

$$\mathbf{x}_t^{d*} = g_i \odot \mathbf{x}_t^d, \quad \mathbf{h}_t^d = \tanh(\mathbf{W}_d \mathbf{x}_t^{d*} + b_d) \qquad (3b)$$

For dot and subtract attention, we will also get h_t^d and h_t^s, respectively. Where $\mathbf{W}_g \in R^{1\times 2d_v}$, $\mathbf{W}_d \in R^{d_v\times 2d_v}$, b_a are weights and bias of our model.

4.3.2. External Aggregation

External aggregation is to fuse all the attention functions. We use a parameter v_i as an input to adaptively fuse two different attention mechanisms.

$$s_j = v^T \tanh(\mathbf{W}_1 h_j^t + \mathbf{W}_2 \mathbf{v}_j)(t = d, s) \qquad (4a)$$

$$a_i = \frac{exp(s_i)}{\sum_{j=(d,s)} exp(s_j)}, \quad \mathbf{x}_t = \sum_{i=(d,s)} a_i \mathbf{h}_t^i \qquad (4b)$$

X={x_1,...,x_N} is the final fused semantic feature. Finally, we feed X into a multilayer perceptron (MLP) classifier for the probability pi of each label in the corresponding task. For all tasks, the objective function is to minimize the following cross entropy:

$$\mathcal{L} = \sum_{i=1}^{N}[y_i \log p_i + (1 - y_i)\log(1 - p_i)] \qquad (5)$$

where y_i denotes a label, in paraphrase detection it is (0, 1) , in natural language inference it is the relation of two sentences of entailment, contradiction, and neutral.

5. EXPERIMENTS AND RESULTS

5.1. Datasets and baselines

Datasets We conduct experiments on 10 sentence matching datasets to evaluate the effectiveness of our method. The GLUE [26] benchmark is a widely-used dataset in thie field, which includes tasks such as sentence pair classification, similarity and paraphrase detection, and natural language inference[1]. We conduct experiments on 6 sentence pair datasets (MRPC, QQP, STS-B, MNLI, RTE, and QNLI) from GLUE. We also conduct experiments on 4 other popular datasets (SNLI, SICK, TwitterURL and Scitail). Furthermore, we tested the robustness of DPM using the Textflint[22] tools.
Baselines To evaluate the effectiveness of our proposed DPM in SSM, we mainly introduce BERT [6] and RoBERTa[7] for comparison. In addition, we also take competitive model transformer[25] without pre-training as baseline. In robustness experiments, we compare the performance of BERT on the robustness test datasets. For simplicity, the compared models are not described in detail here.

5.2. Results and analysis

To evaluate the effectiveness of our method, we test the effectiveness of aggregating DPM in interaction-based and representation-based methods, respectively.

Firstly, we integrate DPM based on interaction-based methods. Table 1 shows the performance of DPM and competitive models on 10 datasets. It can be seen that the effect of non-pre-trained models is significantly worse than pre-trained models. This is mainly because the pre-trained model has more data from learning corpus and powerful information extraction ability. When the backbone model is BERT-base or BERT-large, the average accuracy after integrating DPM is improved by 1.1% and 0.7%, respectively. The results show the effectiveness of our DPM framework on semantic matching tasks. Furthermore, our method outperforms RoBERTa-base by 1.5% and RoBERTa-large by 0.6%, respectively. which demonstrates that DPM can effectively capture the relationship between sentences from different aspects, so that more fine-grained and complex relationships can be exploited. Second, the aggregated representation module can effectively fuse information from different attention modules.

Secondly, to verify the generalization performance of our method, we also test aggregated DPM among representation-based methods. The baseline model has the same settings as Sentence-BERT[27]. It obtains two representations of two texts separately through the encoder, then concatenates the two representations, and finally uses a nonlinear function to distinguish the relationship. The results are shown in Table 2. It can be seen that the representation-based method performs significantly worse than the interaction-based model.This is mainly because interaction-based methods can learn the alignment between phrases in a sentence and can better model sentence-pair relationships. And in the Scitail dataset, due to the small amount of training set data in Scitail, the variance of the model prediction results is large. However, DPM-Net still shows very competitive performance on the Scitail dataset. Furthermore, DPM-Net outperforms vanilla Bert and other competing models on almost all datasets. Those improvements demonstrate the benefit of dual-path modeling for mining semantics.

Overall, consistent conclusions can be drawn from these results. Compared with previous work, our method shows very competitive performance in judging semantic similarity, and the experimental results also confirm our method.

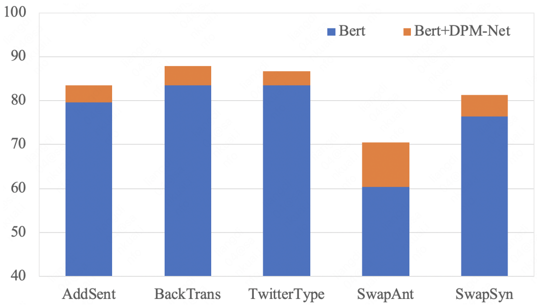


Fig. 3. The robustness experiment with DPM-Net on SNLI dataset .

5.3. Robustness Test Performance

We conducted robustness tests on SNLI dataset. Table 3 lists the accuracy of DMP-Net and baseline model. We can observe that SwapAnt leads to a drop in maximum performance, and our model outperforms Bert nearly 10% on SwapAnt, which indicates that DMP-Net can better handle semantic contradictions caused by antonyms. And the model performance drops to 76.2% on SwapSyn transformation, while DPM-net outperforms BERT by nearly 5% because it requires the model to capture subtle entity differences for correct linguistic inference. In other transformations, DPM-Net still better than baseline, which reflects the advantages of dual path modeling in capturing subtle differences.

Table 3. Results of component ablation experiment.

Model	Quora		QNLI	
	Dev	Test	Dev	Test
DPM-Net	**85.6**	**84.4**	**88.3**	**86.0**
w/o Dot-attention	84.5	83.2	87.1	84.9
w/o Subtract-attention	85.1	83.5	87.3	85.2
w/o Dual Attention	83.9	82.8	86.5	84.7
w/o Internal Fusion	85.3	83.8	87.7	85.6
w/o External Fusion	85.4	83.9	87.9	85.7

5.4. Ablation Study

The experimental results are shown in the table 3. First, remove dual Attention or remove the subcomponents in dual attention, the performance of the model on both datasets is significantly decreased. Which demonstrates the effectiveness of the internal components of the dual attention module. Next, after removing internal fusion or external fusion, the performance of the model decreases by 0.5% and 0.6%, which proves that dynamic aggregation according to different weights can further improve the performance of the model. Overall, due to the effective combination of each component, DPM-Net can adaptively fuse difference features into models and leverage its powerful contextual representation to better inference about semantics.

6. CONCLUSION

In this paper, we propose a novel Dual Path Modeling Network (DPM-Net), which can efficiently aggregate the difference information in sentence pairs. DPM-Net enables the model to learn more fine-grained comparative information and enhances the sensitivity

[1] https://huggingface.co/datasets/glue

of models to subtle differences. Experimental results on 10 public datasets and robustness dataset show that our method can achieve better performance than several strong baselines. Since DPM-Net is an end-to-end training component, it is expected to be applied to other large-scale pre-trained models in the future.

References

[1] N. Madnani, J. Tetreault, and M. Chodorow, "Re-examining machine translation metrics for paraphrase identification," in *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Montréal, Canada: Association for Computational Linguistics, Jun. 2012, pp. 182–190. [Online]. Available: https://aclanthology.org/N12-1019

[2] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," *arXiv preprint arXiv:1508.05326*, 2015.

[3] S. Wang, Y. Lan, Y. Tay, J. Jiang, and J. Liu, "Multi-level head-wise match and aggregation in transformer for textual sequence matching," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 9209–9216.

[4] Q. Chen, X. Zhu, Z. Ling, S. Wei, H. Jiang, and D. Inkpen, "Enhanced lstm for natural language inference," *arXiv preprint arXiv:1609.06038*, 2016.

[5] Y. Tay, L. A. Tuan, and S. C. Hui, "A compare-propagate architecture with alignment factorization for natural language inference," *arXiv preprint arXiv:1801.00102*, vol. 78, p. 154, 2017.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[7] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[8] A. Martins and R. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification," in *International conference on machine learning*. PMLR, 2016, pp. 1614–1623.

[9] X. Qiu and X. Huang, "Convolutional neural tensor network architecture for community-based question answering," in *Twenty-Fourth international joint conference on artificial intelligence*, 2015.

[10] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised learning of universal sentence representations from natural language inference data," *arXiv preprint arXiv:1705.02364*, 2017.

[11] J. Choi, K. M. Yoo, and S.-g. Lee, "Learning to compose task-specific tree structures," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[12] D. Liang, F. Zhang, Q. Zhang, and X.-J. Huang, "Asynchronous deep interaction network for natural language inference," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 2692–2700.

[13] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[14] O. Bodenreider, "The unified medical language system (umls): integrating biomedical terminology," *Nucleic acids research*, vol. 32, no. suppl_1, pp. D267–D270, 2004.

[15] E. Kiperwasser and M. Ballesteros, "Scheduled multi-task learning: From syntax to translation," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 225–240, 2018.

[16] D. Liang, F. Zhang, W. Zhang, Q. Zhang, J. Fu, M. Peng, T. Gui, and X. Huang, "Adaptive multi-attention network incorporating answer information for duplicate question detection," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 95–104.

[17] Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, and X. Zhou, "Semantics-aware bert for language understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 9628–9635.

[18] T. Xia, Y. Wang, Y. Tian, and Y. Chang, "Using prior knowledge to guide bert's attention in semantic textual matching tasks," in *Proceedings of the Web Conference 2021*, 2021, pp. 2466–2475.

[19] J. Bai, Y. Wang, Y. Chen, Y. Yang, J. Bai, J. Yu, and Y. Tong, "Syntax-bert: Improving pre-trained transformers with syntax trees," *arXiv preprint arXiv:2103.04350*, 2021.

[20] S. Wang, D. Liang, J. Song, Y. Li, and W. Wu, "Dabert: Dual attention enhanced bert for semantic matching," in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022, pp. 1645–1654.

[21] J. Song, D. Liang, R. Li, Y. Li, S. Wang, M. Peng, W. Wu, and Y. Yu, "Improving semantic matching through dependency-enhanced pre-trained model with adaptive fusion," *arXiv preprint arXiv:2210.08471*, 2022.

[22] T. Gui, X. Wang, Q. Zhang, Q. Liu, Y. Zou, X. Zhou, R. Zheng, C. Zhang, Q. Wu, J. Ye *et al.*, "Textflint: Unified multilingual robustness evaluation toolkit for natural language processing," *arXiv preprint arXiv:2103.11441*, 2021.

[23] Z. Li, J. Xu, J. Zeng, L. Li, X. Zheng, Q. Zhang, K.-W. Chang, and C.-J. Hsieh, "Searching for an effective defender: Benchmarking defense against adversarial word substitution," *arXiv preprint arXiv:2108.12777*, 2021.

[24] P. Liu, J. Fu, Y. Xiao, W. Yuan, S. Chang, J. Dai, Y. Liu, Z. Ye, Z.-Y. Dou, and G. Neubig, "Explainaboard: An explainable leaderboard for nlp," *arXiv preprint arXiv:2104.06387*, 2021.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[26] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, “Glue: A multi-task benchmark and analysis platform for natural language understanding,” *arXiv preprint arXiv:1804.07461*, 2018.

[27] N. Reimers and I. Gurevych, “Sentence-bert: Sentence embeddings using siamese bert-networks,” *CoRR*, vol. abs/1908.10084, 2019. [Online]. Available: http://arxiv.org/abs/1908.10084