Enhancing Transformer with Sememe Knowledge

Yuhui Zhang[1*], **Chenghao Yang**[2*], **Zhengping Zhou**[1], **Zhiyuan Liu**[1]
[1]Department of Computer Science and Technology, Tsinghua University
[2]Software College, Beihang Unviersity

Abstract

Transformer shows remarkable performances on various natural language processing tasks. However, previous works merely consider Transformer as a data-driven model and fail to incorporate elaborate semantic knowledge. In this paper, we introduce sememe knowledge into Transformer and propose three sememe-enhanced Transformer models. Sememes, by linguistic definition, are the minimum semantic units of language, which can well represent implicit semantic meanings behind words. Our experiments demonstrate that introducing sememe knowledge into Transformer can significantly improve language modeling and downstream tasks. Adversarial test further demonstrates that sememe knowledge can substantially improve model robustness.

1 Introduction

Incorporating syntactic and semantic knowledge into deep neural networks (DNNs) has been an active and controversial topic in natural language processing (NLP). While many state-of-the-art DNNs are treated as purely data-driven models without utilizing explicit linguistic rules (Radford et al., 2018; Devlin et al., 2018), recent works have demonstrated that syntactic information can be beneficial for various NLP tasks (Aharoni and Goldberg, 2017; Strubell et al., 2018).

On the other hand, our work focuses on the semantic aspect, and aims to answer the question: *Can external semantic knowledge be helpful for DNNs?* In response, we incorporate sememe knowledge into DNNs and propose semantically-informed Transformer (Vaswani et al., 2017). Instead of treating words as the minimum semantic units for natural language, some linguists assume that a limited closed set of atomic semantic units (i.e., **sememes**) can be composed to represent the semantic meaning of each word (Bloomfield, 1926). For example, the meaning of "pirate" can be considered as the combination of the meaning of "human", "rob", "guilty" and "waters". Some researchers spend many years annotating words with its sememes and constructing such sememe-based lexical knowledge base (KB). HowNet (Dong and Dong, 2006) is one of the most famous and open-accessed (Qi et al., 2019) KBs that can provide powerful support for models to understand word semantics (Gu et al., 2018; Niu et al., 2017). We adopt HowNet as the sememe KB and try to incorporate sememe knowledge into the input layer, the output layer, and both.

We verify the effectiveness of our models on Language Modeling and three Chinese NLP tasks closely related to word-level and sentence-level semantics (Sec. 3.2). Extensive experiments demonstrate that with the help of sememe knowledge, our model **1)** achieves consistent performance gains on all tasks (Sec. 3.3), and **2)** shows improved robustness against data ablation (Sec. 3.3) and adversarial test (Sec. 3.4). We also **3)** perform a detailed case study to get interpretations on the effectiveness of sememe knowledge (Sec. 3.4).

2 Methodology

In this section, we introduce three ways how we incorporate semantic knowledge into our model architecture: 1) we add aggregated sememe embeddings to the input embeddings to enhance the input representation (**Transformer-SE**); 2) we add sememe prediction as an auxiliary task to enhance the output representation (**Transformer-SP**); 3) the ensemble model (**Transformer-SEP**).

* equal contributions.

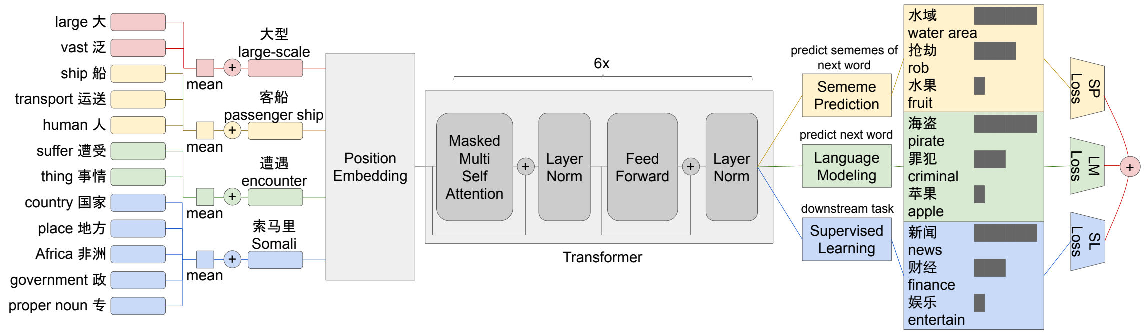


Figure 1: Our proposed model architecture. For each word, we enhance word representation by adding aggregated sememe embeddings. We use multitask learning with three tasks: **sememe prediction** (predicting sememes of next word), **language modeling** (predicting next word) and **supervised learning** (only for downstream tasks).

2.1 Transformer

Transformer was originally proposed by Vaswani et al. (2017) as a machine translation architecture. We use a multi-layer Transformer architecture similar to the setup in Radford et al. (2018), which has been verified effectiveness on multiple NLP tasks. At the input layer, a sequence of words $(w_1, w_2, ..., w_T)$ are embedded as $\mathbf{H}^0 = (\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_T) \in \mathbb{R}^{T \times D}$, where D indicates the hidden size of the model. We then add a positional embedding to inject position information into Transformer. After L residual multi-head self-attention layers with feed-forward connections, we obtain the contextualized sequence embedding $\mathbf{H}^L = (\mathbf{h}_1^L, \mathbf{h}_2^L, ..., \mathbf{h}_T^L) \in \mathbb{R}^{T \times D}$.

2.2 Aggregated Sememe Embeddings

For each word w, Transformer-SE considers all of its sememes and enhances word representation by adding its average sememe embeddings to word embedding. Formally, we have:

$$\tilde{\mathbf{w}} = \frac{1}{n_w} \sum_{s \in S(w)} \mathbf{x}_s + \mathbf{w}$$

where $S(w)$ refers to the sememe set associated with word w with the size n_w, $\mathbf{x}_s$ refers to the embedding of the sememe s, $\mathbf{w}$ refers to the embedding of word w and $\tilde{\mathbf{w}}$ refers to the sememe-enhanced word embedding. Sememe-enhanced representation $\tilde{\mathbf{w}}$ is fed directly to Transformer.

The Transformer-SE model complies with the linguistic assumption that implicit word semantics can be composed by a limited set of sememes. Also, because sememe embeddings are shared among words, latent semantic correlations between words can be well encoded. While the Transformer-SE model seems pretty straightforward, we use this model to probe the effectiveness of sememe knowledge.

2.3 Sememe Prediction Auxiliary Task

Sememe prediction task (Xie et al., 2017) aims to predict sememes for a given word and can be formulated as a multilabel classification task. This task challenges the model with the capability of sememe knowledge incorporation, which is closely related to the understanding of word semantics. Inspired by multitask learning (Ruder, 2017), we add sememe prediction task in addition to language modeling task for Transformer-SP.

Given a word w with its output contextualized embedding $\mathbf{h}^L$ from Transformer, we estimate the probability of word w associated with sememe s as $p(w, s) = \sigma(\mathbf{W}\mathbf{h}^L + b)$, where $\mathbf{W}$ and b are the weight and bias of the dense layer, σ is the sigmoid activation function. We then calculate the binary cross-entropy loss of sememe prediction $\mathcal{L}_{SP}$ as:

$$\mathcal{L}_{SP} = -\frac{1}{n} \sum_{s \in S} g(w, s) \log(p(w, s))) \\ +(1 - g(w, s)) \log(1 - p(w, s))$$

where S refers to the overall sememe set with the size n, $g(w, s)$ is a binary variable indicating whether word w is associated with sememe s in the gold label. Finally, we formulate the loss as:

$$\mathcal{L}_{PRE} = \mathcal{L}_{LM} + \mathcal{L}_{SP}$$
$$\mathcal{L} = \mathcal{L}_{SL} + \rho \mathcal{L}_{PRE}$$

where $\mathcal{L}_{LM}$ and $\mathcal{L}_{SL}$ are the conventional negative log-likelihood language modeling loss and downstream supervised learning loss. $\mathcal{L}_{PRE}$ is the loss optimized during pretraining, while $\mathcal{L}$ is the loss optimized during supervised training for downstream tasks, ρ serves as a coefficient to control the strength of $\mathcal{L}_{PRE}$ during supervised learning.

2.4 Ensemble Model

Transformer-SE and Transformer-SP are designed based on different ideas. Transformer-SE can well inform sememe knowledge to all self-attention layers, while Transformer-SP utilizes additional training signals through the back-propagation process. To combine the advantages of these two models, we propose an ensemble model named Transformer-SEP. Transformer-SEP incorporates sememe knowledge into the input layer by adding aggregated sememe embeddings and performs sememe prediction auxiliary task in the output layer.

3 Experiments

We experiment across a diverse set of five benchmark NLP tasks and demonstrate the effectiveness of adding sememe knowledge.

3.1 Experimental Setup

We use 6-layer 8-head Transformer with the hidden size of 768 and feedforward size of 2048. We set both word embedding and sememe embedding size as 768. We use batch size of 32 and set dropout rate as 0.2 to alleviate overfitting. The vocabulary size is 39,770 and the total number of sememes is 2,100. We truncate the sequence length to 128 for pretraining and supervised learning. When performing supervised training, we set the coefficient ρ to be 0.5. Embeddings are tied for words and sememes to speed up convergence. We clip gradients less than 2 and use Noam optimizer with 0.001 learning rate and 8000 warmup steps. For downstream tasks, we use the best pretrained model from language modeling to initialize.

3.2 Tasks and Datasets

Language Modeling Language modeling on a large corpus provides additional training signals for supervised downstream tasks. We use perplexity (PPL) to measure the performance of language model. Lower PPL indicates better performance. We pretrain the language model on the People's Daily corpus, which contains $\sim$ 15M words.

Headline Categorization Automatic and accurate news categorization is essential for recommendation systems. We use NLPCC 2017 news headline categorization dataset (Qiu et al., 2017), which contains 156,000 news for training and 36,000 news for validation, divided into 18 categories including finance, society, game, etc. We use accuracy (ACC) to measure the performance.

Sentiment Classification Sentiment classification is a useful task for emoticon recommendation, depression detection, etc. We use NLPCC 2013 weibo sentiment detection dataset and conduct experiments on sentence-level sentiment classification. The dataset includes 7 different sentiment genres. We remove sentences without any sentiment and resplit the data to 8,225 / 997 / 1,020 for training, validation, test, respectively.

Semantic Matching Semantic matching is fundamental for question answering, which aims to match the input question to similar questions in an existing database. We use LCQMC (Liu et al., 2018) dataset for this task, which contains 238,766 / 8,802 / 12,500 training, validation, test data, respectively. For each pair of questions, we concatenate them with a special token for classification.

Sememe Prediction Predicting sememes for given words by its definitions is important for the HowNet extension (Xie et al., 2017). The definitions are extracted from the Contemporary Chinese Dictionary and the sememes of target words are masked for fair comparison. We create a dataset containing 41,081 / 5,135 / 5,136 word-definition pairs for training, validation and test.

3.3 Overall Performance

From Table 1, we observe that simply adding sememe embedding (i.e., Transformer-SE) can lead to significant improvements over all tasks. These tasks challenge models on the capability of modeling word-level semantics and sentence-level semantics, which demonstrates that sememe knowledge can provide beneficial semantic information for Transformer. The improvement of Transformer-SP is rather less, which may due to the difficulty of predicting new knowledge without previous knowledge. Transformer-SEP achieves further improvements over Transformer-SE. The additional improvement can be interpreted as combining the advantages of these two models.

We also compare sememe decomposition to character decomposition for our best model (i.e., with aggregated character embedding and character prediction auxiliary task) (Table 1). We ob-

Task Metric	Language Modeling PPL	Headline Categorization ACC (%)	Sentiment Classification ACC (%)	Semantic Matching ACC (%)	Sememe Prediction MAP (%)
Transformer	49.01	71.5	52.7	81.2	40.1
Transformer-SE	47.37	72.6	53.7	82.6	52.1
Transformer-SP	49.14	72.3	53.0	81.8	40.3
Transformer-SEP	**46.53**	**72.6**	**54.9**	**83.3**	**52.8**
+ Sememe2Char	48.90	72.3	52.2	81.2	-

Table 1: Experimental results on different tasks. **Transformer**, **Transformer-SE**, **Transformer-SP** and **Transformer-SEP** refers to the vanilla Transformer model (base), Transformer with aggregated sememe embeddings, Transformer with sememe prediction auxiliary task and the ensemble model, respectively. We also compare sememe decomposition to character decomposition for our best model and demonstrate advantages of our methods.

Replace	Semantic Matching			Sentiment Classification			Headline Categorization		
	#Count	Base	Ours	#Count	Base	Ours	#Count	Base	Ours
-	0	0.0	0.0	0	0.0	0.0	0	0.0	0.0
Noun.	30,858	18.0	**15.4**(-14%)	2,313	14.1	**11.8**(-16%)	168,516	14.8	**13.4**(-10%)
Adj.	6,498	16.7	**14.8**(-11%)	1,143	20.4	**16.9**(-17%)	54,054	9.4	9.5(+1%)
Adv.	3,306	16.1	**14.1**(-12%)	1,803	14.0	**12.3**(-12%)	65,136	8.5	**8.0**(-6%)
ALL	40,662	17.6	**15.2**(-14%)	5,259	15.4	**13.1**(-15%)	287,706	12.4	**11.4**(-8%)

Table 2: Adversarial test for base model and our model (i.e., Transformer v.s. Transformer-SEP). We generate adversarial data by replacing nouns, adjectives and adverbs in data that both models can correctly predict. We report **error rate** (lower the better) categorized by part-of-speech and the number of generated adversarial examples.

serve significant performance drops over all tasks and demonstrate decomposing word into sememes are much more efficient. We further perform data ablation study and observe overall consistent improvements for downstream tasks over different amounts of training data (shown in Appendix).

3.4 Adversarial Test and Case Study

Recent research has demonstrated that DNNs are vulnerable to adversarial examples (Goodfellow et al., 2014; Jia and Liang, 2017; Alzantot et al., 2018). We generate adversarial examples by replacing similar nouns, adjectives and adverbs in the data that both Transformer and Transformer-SEP can predict correctly. We calculate the word similarity based on the novel Cilin metric (Tian and Zhao, 2010) and we use THULAC (Sun et al., 2016) for part-of-speech (POS) tagging. The basic intuition is that these words are generally more informative for prediction. In semantic matching task, we only replace shared words in each pair of sentences, which ensures semantic consistency.

We report the adversarial test error rate categorized by POS in Table 2. Sememe-enhanced Transformer-SEP achieves consistent improvement over the vanilla Transformer. It is also intuitive that, in headline categorization and semantic matching, the largest performance drops are observed by replacing nouns, and in contrast sentiment classification is more sensitive to adjectives.

奸商（骗子）如何有工作牌在行李大厅里明目张胆行骗?

How do the **profiteers** **(cheaters)** have staff cards and blatantly cheat in the baggage hall?

有罪 guilty **人 human** **欺骗 deceive** **商业 commerce**

有罪 guilty **人 human** **骗 cheat**

Table 3: Case study for adversarial test. The **original word** with its sememes are colored in blue, while the **replaced word** with its sememes are colored in red.

We further perform case study to get a better interpretation of why sememe knowledge can improve model robustness to adversarial attack (Table 3). We show an example that Transformer-SEP can predict correctly but get wrong for Transformer. More details are shown in Appendix.

4 Conclusion

In this work, we introduce sememe knowledge into Transformer and achieve significant improvements over multiple tasks, verifying the effectiveness of explicit semantic knowledge for DNNs. We further demonstrate the robustness of our methods via data ablation study, adversarial test and case study. In the future, we will explore more ways to leverage semantic knowledge and generate different adversarial examples for evaluation.

References

Roee Aharoni and Yoav Goldberg. 2017. Towards string-to-tree neural machine translation. *arXiv preprint arXiv:1704.04743*.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. *arXiv preprint arXiv:1804.07998*.

Leonard Bloomfield. 1926. A set of postulates for the science of language. *Language*, 2(3).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Zhendong Dong and Qiang Dong. 2006. *Hownet and the computation of meaning (with Cd-rom)*. World Scientific.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Yihong Gu, Jun Yan, Hao Zhu, Zhiyuan Liu, Ruobing Xie, Maosong Sun, Fen Lin, and Leyu Lin. 2018. Language modeling with sparse product of sememe experts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4642–4651.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. *arXiv preprint arXiv:1707.07328*.

Xin Liu, Qingcai Chen, Chong Deng, Huajun Zeng, Jing Chen, Dongfang Li, and Buzhou Tang. 2018. Lcqmc: A large-scale chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1952–1962.

Yilin Niu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2017. Improved word representation learning with sememes. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2049–2058.

Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Qiang Dong, Maosong Sun, and Zhendong Dong. 2019. Openhownet: An open sememe-based lexical knowledge base. *arXiv preprint arXiv:1901.09957*.

Xipeng Qiu, Jingjing Gong, and Xuanjing Huang. 2017. Overview of the nlpcc 2017 shared task: Chinese news headline categorization. In *arXiv:1706.02883v1*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

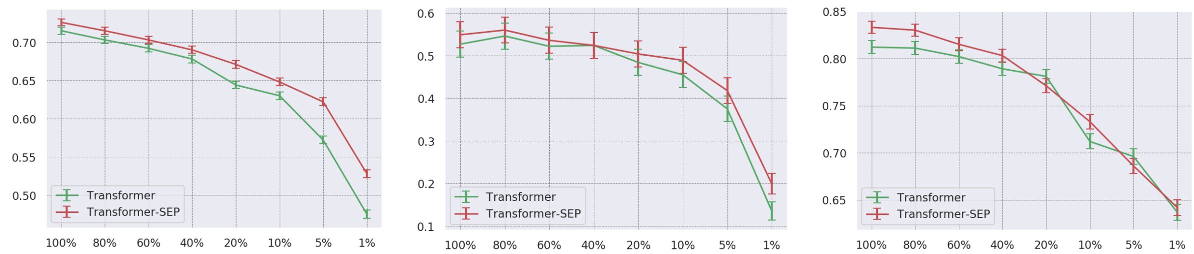
Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5027–5038.

Maosong Sun, Xinxiong Chen, Kaixu Zhang, Zhipeng Guo, and Zhiyuan Liu. 2016. Thulac: An efficient lexical analyzer for chinese. Technical report, Technical Report. Technical Report.

Jiu-le Tian and Wei Zhao. 2010. Words similarity algorithm based on tongyici cilin in semantic web adaptive learning system. *Journal of Jilin University(Information Science Edition)*, 28(6):602–608.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Ruobing Xie, Xingchi Yuan, Zhiyuan Liu, and Maosong Sun. 2017. Lexical sememe prediction via word embeddings and matrix factorization. In *IJCAI*, pages 4200–4206.



Headline Categorization Sentiment Classification Semantic Matching

Figure 2: Performance of Transformer-SEP and Transformer with different amounts of training data. X-axis: Percent of supervised training data. Y-axis: ACC. The error bars indicate the 95% confidence interval.

A Supplementary Materials

A.1 Data Ablation Study

This experiment studies the robustness of Transformer-SEP to different amounts of supervised training data, which are randomly sampled from the original training set. A comparison against Transformer is shown in Figure 2. Generally, Transformer-SEP achieves consistent improvements over Transformer, which indicates that incorporating external sememe knowledge could benefit model robustness when faced with limited training data.

It is also worth noting that, Transformer-SEP achieves notable improvements on Headline Categorization and Sentiment Classification, yet only comparable performance on Semantic Matching. This is probably due to the fact that Headline Categorization depends mainly on **word-level semantics** (e.g. the word *football* strongly indicates *sport*), while Semantic Matching focuses mostly on **sentence-level semantics** (e.g. *what's football?* $\neq$ *is it a football?*), and Sentiment Classification is closer to a combination of both (e.g. the word *awesome* strongly indicates *like*, yet *Had it not rained, it would have been an awesome day* is closer to *sad*).

Hence, when the amount of training data decreases, the amount of unseen words in test set increases. However, similar (but different) words would share similar sememes, and hence our sememe-informed model would strongly outperform the baseline in those scenarios. Specifically, when training data is limited, the more a task depends on **word-level semantics**, the larger improvement could be achieved by incorporating sememe knowledge.

A.2 Adversarial Case Study

Deep neural networks are vulnerable to adversarial examples (Alzantot et al., 2018). To evaluate the robustness of our models, we generate adversarial examples by replacing similar nouns, adjectives and adverbs in the data that both Transformer and Transformer-SEP can predict correctly. We show a set of adversarial examples that Transformer-SEP still predicts successfully, yet get wrong answer from Transformer in Table 4.

From Table 4, we can easily interpret why sememe knowledge can enhance word representation and resist adversarial attack. For example, word "cheater" and "profitter" share the same sememes "guilty" and "human", and share similar sememe "deceive" and "cheat". These sememe knowledge can propagate through all self-attention layers, and thus help the model to resist attack if we replace "cheaters" to "profitters" in sentence "How do the profiteers (cheaters) have staff cards and blatantly cheat in the baggage hall?".

We also get intuition on how both methods (i.e. aggregated sememe embeddings and sememe prediction auxiliary task) boost model performance. As for the former one, word representation can be enhanced as discussed above and help model to better understand unseen concepts; As for the latter one, the model learns the relatively easy sememe prediction task as well as the more difficult language modeling task (with about 20 times larger vocabulary size). For instance, if a sentence starts with "How to choose", we would suggest the next word is a kind of "tool", that is exactly what sememe prediction does and thereby helps the model to gain better semantic representation for sentences.

Task	Input	Ours	Base
Sentiment Classification	奸商（骗子） 如何有工作牌在行李大厅里明目张胆行骗? How do the **profiteers (cheaters)** have staff cards and blatantly cheat in the baggage hall? **有罪 guilty** **人 human** **欺骗 deceive** **商业 commerce** **有罪 guilty** **人 human** **骗 cheat**	disgust	surprise
	吓人（可怕），中药比西药更不安全。 **Frightful (Fearful)**, Chinese medicine is less safe than Western medicine. **能 able** **促使 urge** **害怕 fear** **能 able** **促使 urge** **害怕 fear**	fear	disgust
Headline Categorization	转载一个成方（秘方），主治一切骨折，据说一剂见效 We republish a **set prescription (secret prescription)** , which mainly treats all kinds of fractures, and is said to be effective with only one dose. **医 medical** **药物 medicine** **准备 prepare** **文书 document** **命令 order** **医 medical** **药物 medicine** **有效 effective** **医治 doctor** **全 all** **方法 method** **疾病 disease**	regimen	essay
	他是三征高句丽的强将（猛将），最后死于一群无赖之手 He was a **good general (valiant general)** that attacked Goguryeo for three times, yet was killed by a group of rogues. **人 human** **军 military** **官 official** **人 human** **军 military** **官 official** **军队 army** **勇 brave** **争斗 fight**	history	story
Semantic Matching	A. 如何选择大哥大（手机）？ A. How to choose **hand phone (mobile phone)**? B. 怎么选择大哥大（手机）？ B. What is the way to choose **hand phone (cell phone)**? **携带 bring** **能 able** **用具 tool** **交流 communicate** **样式值 PatternValue** **携带 bring** **能 able** **用具 tool** **交流 communicate** **样式值 PatternValue**	same	different
	A. 初中生（男生） 暗恋女生会有什么表现? A. What performance will **junior high school students (boy students)** have if they secretly love a girl? B. 初中生（男生） 暗恋女生表现是什么? B. What is the performance of **junior high school students (boy students)** if they secretly love a girl? **学习 study** **教 teach** **场所 InstitutePlace** **人 human** **教育 education** **中等 intermediate** **学习 study** **教 teach** **场所 InstitutePlace** **人 human** **教育 education** **初等 elementary** **男 male**	same	different

Table 4: Case Study for adversarial test. The **original words** are shown in parenthesis and colored in blue, while the **replaced words** (similar words calculated by Cilin (Tian and Zhao, 2010)) are colored in red. Both the base model and our model (i.e. Transformer v.s. Transformer-SEP) predict correctly on sentences with the original words, yet only ours succeed on the sentences with the replaced words. We show **sememes for original words** and **sememes for replaced words** in blue and red color boxes respectively. *Best viewed in color.*