

# W2CL: A Multi-Task Learning Approach to Improve Domain-Specific Sentence Classification through Word Classification and Contrastive Learning

Anonymous

Anonymous

**Abstract.** Sentence classification task plays a crucial role in various NLP tasks. Recent studies have shown that contrastive learning can enhance the representational capability of Pre-trained Language Models (PLMs) and that different methods for constructing positive and negative samples can be applied to various downstream application scenarios. Therefore, in this study, we propose **W2CL**, a novel multi-task learning framework based on **W**ord **C**lassification and **C**ontrastive **L**earning, aimed at integrating domain knowledge extracted by ChatGPT from raw corporas into PLMs and improving the performance of PLMs in domain-specific sentence classification tasks. Contrastive learning assists the model in gradually learning the semantic similarity and contextual relevance between words during the training process to enhance its ability to understand text. Word classification provides additional contextual understanding, thereby improving the model's ability to differentiate between different classes within the specific domain. Experiments demonstrate that our multi-task approach significantly outperforms other methods, leading to substantial improvements in domain-specific sentence classification performance. This framework offers a robust solution for adapting general-purpose language models to specialized domains, ensuring better performance and generalization in various domain applications.

**Keywords:** Sentence Classification · Contrastive Learning · Multi-task Learning.

## 1 Introduction

In recent years, mainstream sentence classification tasks have largely relied on deep representation learning methods, especially PLMs such as BERT [3], RoBERTa [11], and T5 [18]. Although the vector representations from these models capture richer textual features compared to traditional methods, simple fine-tuning is insufficient for integrating domain-specific knowledge into language models for domain-specific sentence classification tasks. Therefore, we propose the W2CL framework to effectively incorporate domain knowledge into the language models to improve the performance of these models in domain-specific tasks.

Numerous studies have demonstrated that integrating effective knowledge with PLMs can enhance the performance of PLMs on certain downstream tasks. LIBERT [8] adds a lexical relationship classification task to the original BERT pre-training tasks to help the model acquire richer semantic information. SenseBERT [9] adds a part-of-speech layer to the BERT model, that is, it adds part-of-speech information of words (such as noun.food and noun.state, etc.) to the original input, and uses the representation vector after integrating part-of-speech information for masking tasks and part-of-speech classification tasks. SKEP [22] integrates sentiment knowledge (sentiment words, sentiment polarity, and aspect-sentiment pairs) into the model by designing three types of sentiment analysis tasks. Sentiprompt [10] integrates sentiment knowledge by constructing different paradigm templates and masking aspects, polarities, and opinions in the templates, and designing tasks to predict the masked tokens. LET [13] integrates all classification definitions of HowNet entities into the input, uses the original input and classification information for masking tasks, and finally integrates the knowledge contained in HowNet into the model. KEAR [27] directly faces the multiple-choice task, integrates the knowledge relationship of questions and options, the dictionary’s definition knowledge of questions and options, and the knowledge of annotated training samples into the model, improving the model’s performance in the commonsense knowledge question answering task.

The aforementioned methods can enhance the performance of models in various downstream tasks. However, these methods have two issues: First, knowledge cannot be extracted from the raw corporas, which means that acquiring knowledge is challenging; Second, the training frameworks of the aforementioned methods are complex, requiring intricate training procedures to match the corresponding forms of knowledge.

With the development of large language models (LLMs) [3, 5, 7, 11, 19, 21, 23, 29], particularly ChatGPT [17], which has attracted significant attention from NLP researchers, some studies [12, 14–16, 20] have shown that ChatGPT has demonstrated impressive performance, outperforming many models even in zero-shot settings. As a result, for issue 1, we utilize ChatGPT to acquire domain knowledge from raw corporas, a process that is both simple and results in high-quality knowledge.

Meanwhile, some studies have demonstrated that contrastive learning has superior capabilities in the field of NLP. SimCSE [4] uses the randomness of the dropout layer to construct positive samples, treating other samples in the same batch as negative samples. Building on SimCSE, ESimCSE [24] uses random word repetition to construct positive samples. PromptBERT [6] applies multiple templates that do not affect semantics within the same sample, treating samples reconstructed with different templates as positive samples. BGE [26] treats the question and its corresponding answer as positive samples, and other samples in the same batch as negative samples. For domain-specific tasks, domain vocabulary contains more domain-specific information compared to sentences. Therefore, for issue 2, we propose word-level contrastive learning and word classification, which together form the proposed W2CL framework.

To better validate the effectiveness of our proposed method, we need to test it on datasets from multiple domains. However, sentence classification datasets in the Chinese domain are not sufficiently abundant. Therefore, we will contribute our proprietary domain-specific datasets for future research and development, which include text retrieval datasets from four domains and sentence classification datasets from two domains.

Overall, our contributions can be summarized from four perspectives. 1) We propose a novel framework called W2CL, which can better transfer general models to specific domains. 2) We propose a novel and simple general method for knowledge extraction and integration. The knowledge extraction method based on ChatGPT and the knowledge integration method based on word-level contrastive learning and word classification are both effective and scalable. 3) Our experimental results demonstrate that the proposed method can enhance the performance of various language models across multiple domains and tasks, indicating that the W2CL framework is both effective and generalizable. 4) For domain-specific tasks, we contribute six task datasets to support future research and development.

## 2 The Proposed Method

This section introduces the detailed methodology of the our proposed W2CL framework, which mainly consists of two parts: domain knowledge extraction with ChatGPT and knowledge integration based on multi-task learning.

### 2.1 Domain Knowledge Extraction with ChatGPT

In the domain knowledge acquisition phase, we use ChatGPT as a knowledge extractor to extract the necessary domain knowledge for multi-task learning from raw text corporas which can be easily obtained from online resources like Baidu Zhidao. This step can be divided into three stages, as shown in Figure 1.

In Step 1, we first use the Jieba tool to segment the raw text corpus and then compile a list of frequently occurring words. ChatGPT is then employed to filter out domain-specific vocabulary from this list, as illustrated in Figure 1.a, and the prompt1 shown in the Figure 1 is "I currently have several terms. You will categorize all the terms I provide into levels of relevance: high, medium, and low. This relevance level pertains to the XX domain". Finally, extract the terms with high relevance to compile the final domain-specific vocabulary list.

In Step 2, ChatGPT is used to categorize the domain-specific vocabulary obtained in Step 1 into relevant categories within the domain, and the prompt2 is "Assume you are now an expert in the XX domain. Please classify the terms I provide within the subcategories of this domain.". This step allows us to obtain the initial version of domain-specific classification knowledge.

In Step 3, we manually consolidate all categories obtained in Step 2 to create a final comprehensive category list that covers all terms. Finally, we provide ChatGPT with this finalized category list to reclassify the vocabulary, producing

the ultimate domain knowledge, and the prompt3 is "Assume you are an expert in the XX domain. I will provide you with several terms related to this domain, along with a list of categories: [Categories]. I need you to assign each of my terms to a category from the provided list. If a term cannot be assigned to any of the existing categories, please mark it as None. The required format for the response is 'Term: Category'".

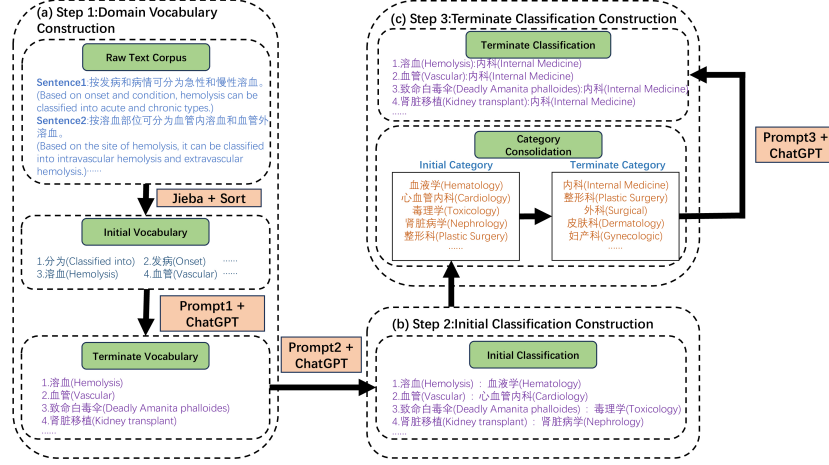


Fig. 1. Knowledge Extraction Framework

## 2.2 Domain Knowledge Integration

Our proposed knowledge integration method is based on Whole Word Masking(WWM)[2] and incorporates two additional training tasks: Word Classification and Contrastive Learning. The details about WWM will be introduced in the BASELINES section of Chapter 3. The following sections will introduce other two tasks individually.

**Word Classification.** Given a sentence  $\{x_1, x_2, x_3, [\text{MASK}], x_5, [\text{MASK}], x_7\}$  where  $x_i$  means one word which may cover several tokens and a list of categories for the masked words  $\{C_1, C_2\}$ . Word Classification adds a classification layer to the original model architecture. The classification loss function is then computed as shown in Equation 1:

$$L(\mathbf{y}, \hat{\mathbf{y}}) = - \sum_{i=1}^C (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)) \quad (1)$$

- $\mathbf{y} = [y_1, y_2, \dots, y_C]$ : The multi-label binary vector of true labels, where  $C$  is the number of classes. Each  $y_i$  represents the true label for class  $i$  (1 indicates the presence of the class, 0 indicates the absence of the class).
- $\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_C]$ : The predicted probability distribution vector from the model, where each  $\hat{y}_i$  represents the predicted probability for class  $i$ , with values in the range  $[0, 1]$ .
- $L(\mathbf{y}, \hat{\mathbf{y}})$ : The loss function value for the multi-label classification task.

**Contrastive Learning.** Given a sentence  $\{x_1, x_2, x_3, [\text{MASK}], x_5, [\text{MASK}], x_7\}$  where  $x_i$  means one word which may cover several tokens, and the actual words masked by  $[\text{MASK}]$   $\{x_4, x_6\}$ . Our proposed contrastive learning method uses the actual word corresponding to the current  $[\text{MASK}]$  position as the positive sample for the current  $[\text{MASK}]$ . All the positive samples of other  $[\text{MASK}]$  tokens from other sentences in the same batch are used as the negative samples for the current  $[\text{MASK}]$ . The specific calculation method is shown in Equation 2:

$$L(\mathbf{t}, \mathbf{t}') = -\frac{1}{N} \sum_{i=1}^N \log \frac{e^{\text{sim}(f(\mathbf{t}_i) \cdot f(\mathbf{t}'_i))/\tau}}{\sum_{j=1}^N e^{\text{sim}(f(\mathbf{t}_i) \cdot f(\mathbf{t}'_j))/\tau}} \quad (2)$$

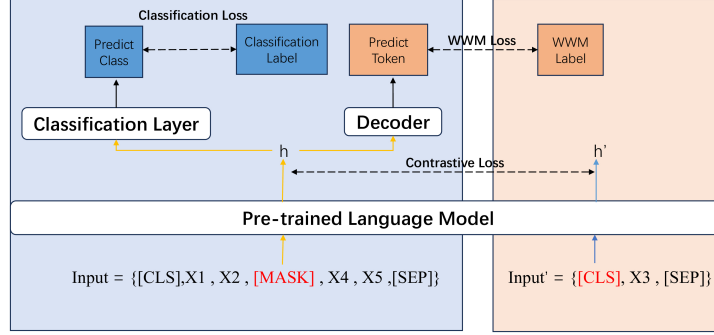
- $\mathbf{t} = [t_1, t_2, \dots, t_N]$ :  $N$  is the number of  $[\text{MASK}]$  in one batch. Each  $f(t_i)$  represents the representation of  $i_{st}$   $[\text{MASK}]$  token in the batch.
- $\mathbf{t}' = [t'_1, t'_2, \dots, t'_N]$ :  $f(t'_i)$  represents the representation of the true label  $t'_i$  corresponding to the  $i_{st}$   $[\text{MASK}]$  token, which is the positive sample for  $f(t_i)$ .
- $L(\mathbf{t}, \mathbf{t}')$ : The loss function value for the contrastive learning task.
- $\text{sim}(\mathbf{h1} \cdot \mathbf{h2})$ :  $\text{sim}(\mathbf{h1} \cdot \mathbf{h2})$  is the cosine similarity  $\frac{\mathbf{h1}^T \mathbf{h2}}{\|\mathbf{h1}\| \cdot \|\mathbf{h2}\|}$ .
- $\tau$ : is a temperature hyperparameter, with a default value of 0.01.

**Training Framework.** The overall training objective comprises three parts:

$$L_{\text{overall}} = L_{\text{WWM}} + L_{\text{WC}} + L_{\text{CL}} \quad (3)$$

where  $L_{\text{WWM}}$  is the loss of the WWM task,  $L_{\text{WC}}$  is calculated by equation 1, and  $L_{\text{CL}}$  is calculated by equation 2.

The training framework is depicted in Figure 2. Firstly, we obtain the loss( $L_{\text{WWM}}$ ) for the WWM task, which aims to help the model learn the meanings, usage, and contextual relationships of vocabulary, facilitating the model’s adaptation to the specific domain context. Secondly, the loss( $L_{\text{CL}}$ ) from contrastive learning can assist the model in gradually learning the semantic similarity and contextual relevance between words during the training process, thereby enhancing its ability to understand text. Finally, the loss( $L_{\text{WC}}$ ) from word classification provides additional contextual understanding and domain knowledge, thereby improving the model’s ability to differentiate between different classes within the specific domain.



**Fig. 2.** Training Framework

### 3 Experiments

In this chapter, we validate the proposed W2CL method on sentence classification tasks across four domains. Additionally, we demonstrate the effectiveness of this method in domain transfer tasks.

#### 3.1 Dataset and Experiment Details

We evaluated the proposed multi-task learning method on four sentence classification datasets, covering the domains of automotive, legal, finance, and medicine. For the transfer learning tasks, we assessed the method on four text retrieval datasets from the same domains. For the sentence classification task datasets, the automotive and financial domains are the datasets we contributed, while the legal and medical domains use public datasets. The text retrieval task datasets are all contributed by us. Below, we provide a brief introduction to these datasets.

**Sentence Classification Dataset.** The evaluation sets for sentence classification task consist of four datasets: Automatic Sentence Classification Corpora (ASCC), CAIL [25], CHIP-CTC [28], and Finance Sentence Classification Corpora (FSCC). ASCC pertains to the automotive domain and includes 6174 training instances and 1000 test instances, with coverage of 10 categories. CAIL is related to the legal domain and comprises 154592 training instances and 49639 test instances, encompassing 134 categories. CHIP-CTC pertains to the medical domain, with 24516 training instances and 6128 test instances, covering 44 categories. FSCC is a finance-related dataset, consisting of 10346 training instances and 3332 test instances, and spans 14 categories.

**Text Retrieval Dataset.** The text retrieval datasets comprise four collections: Automatic Text Retrieval Corpora (ATRC), Legal Text Retrieval Corpora (LTRC), Medical Text Retrieval Corpora (MTRC), and Finance Text Retrieval Corpora (FTRC). These datasets correspond to the automotive, legal, medical, and financial domains, respectively. Each dataset contains 1000, 232, 3520, and 1953 retrieval test texts, with a retrieval text space of 7175, 5232, 15000, and 12000 texts accordingly.

**Experiment Details.** Our model begins with the checkpoint of BERT/RoBERTa model, and we utilize the [CLS] token representation as the sentence representation. During training, we employ an Adam optimizer with a batch size of 16. The learning rate for the sentence representation model is set to  $5e-5$ . During the fine-tuning stage, we modified the learning rate to  $3e-5$  based on the training hyperparameters. Additionally, we set the seed to 0, 1, and 2, respectively, and calculated the average experimental data metrics for these three scenarios.

### 3.2 BASELINES

We validated the effectiveness of our proposed method by comparing it against three baseline approaches: Whole Word Masking(WWM), SimCSE, and SSCL[1], using BERT and RoBERTa as basic models.

**WWM.** WWM modifies the random masking task by changing the masking unit from token to whole word, leveraging contextual information to predict the masked word. For instance, in WWM, both "play" and "#ing" would be masked together. We selected WWM as a baseline method instead of random masking because, for domain-specific tasks, domain-specific word tends to contain more domain-relevant information.

**SimCSE.** SimCSE constructs positive samples based on the randomness of dropout and uses in-batch sampling to create negative samples. The model is trained by generating a loss value through the contrast between the original samples and the positive and negative samples.

**SSCL.** SSCL enhances the construction of negative samples by leveraging the principles of SimCSE. Specifically, SSCL employs hidden representations from intermediate layers of PLMs as negative samples. The objective is to ensure that the final sentence representations are distinctly separated from these intermediate representations.

### 3.3 Experiment Results

In this chapter, we will present the experimental results for the sentence classification and retrieval tasks. F1 score and Mean Reciprocal Rank(MRR) are used as evaluation metrics for these two tasks, respectively.

**Sentence Classification Task.** Table 1 presents the performance of various methods on the sentence classification task, with F1 score as the evaluation metric. From the experimental results, it can be observed that the proposed W2CL method outperforms other methods. Compared to the WWM method, W2CL incorporates contrastive learning task and word classification task on top of WWM. In comparison to SimCSE and SSCL methods, W2CL modifies the construction of positive and negative samples for contrastive learning and introduces word classification task. Based on the experimental results, the following conclusions can be drawn: 1) Domain knowledge constructed based on ChatGPT is effective; 2) The proposed W2CL method effectively integrates domain knowledge into the model; 3) The proposed W2CL method outperforms other methods across different baseline models and domains. This indicates that W2CL can serve as a universal method, adaptable to any model as the basic model and transferable across various domains.

**Table 1.** Evaluation performance on the sentence classification tasks. The evaluation metric is F1 score(%).

Model	ASCC	CAIL	CHIP-CTC	FSCC	Avg.
<i>BERT Version</i>					
BERT	92.48	65.58	82.91	85.82	81.70
WWM-BERT	94.08	66.62	83.52	85.91	82.53
SimCSE-BERT	93.70	66.68	83.64	85.81	82.46
SSCL-BERT	93.76	66.43	83.56	85.67	82.36
W2CL-BERT(Ours)	<b>94.93</b>	<b>66.83</b>	<b>84.06</b>	<b>86.08</b>	<b>82.98</b>
<i>RoBERTa Version</i>					
RoBERTa	93.93	67.38	83.91	86.10	82.83
WWM-RoBERTa	94.26	67.46	83.59	86.11	82.86
SimCSE-RoBERTa	94.14	67.43	83.92	85.83	82.83
SSCL-RoBERTa	94.41	67.18	83.56	86.27	82.86
W2CL-RoBERTa(Ours)	<b>94.84</b>	<b>67.71</b>	<b>83.98</b>	<b>86.53</b>	<b>83.27</b>

**Transfer Task.** In this section, we evaluate the generalization ability of the proposed W2CL method on text retrieval tasks across four domains. The objective of this task is to rank the candidate retrieval texts based on their similarity to the given retrieval text. For this task, we directly utilize the representation vector of the [CLS] token, without the need for fine-tuning.

The experimental results are shown in Table 2. From the table, it can be observed that the proposed W2CL method achieves superior MRR scores on most datasets. This result indicates that the W2CL method exhibits better generalization ability compared to other methods in the study. When using RoBERTa as the basic model, the average MRR of WWM and SimCSE is lower than that



of RoBERTa. As shown in the table, this is mainly because the MRR of WWM and SimCSE is lower than RoBERTa on the automotive domain dataset, leading to a lower average MRR. Since there is no fine-tuning phase in this task, we guess that this is primarily due to the randomness of the dataset and pre-training.

**Table 2.** Evaluation performance on the text retrieval tasks. The evaluation metric is MRR score(%).

Model	ATRC	LTRC	MTRC	FTRC	Avg.
<i>BERT Version</i>					
BERT	5.22	63.36	53.86	48.91	42.84
WWM-BERT	63.33	76.93	78.14	62.81	70.30
SimCSE-BERT	68.48	84.49	86.90	66.48	76.59
SSCL-BERT	32.91	85.85	79.70	79.44	69.48
W2CL-BERT(Ours)	<b>72.30</b>	<b>93.54</b>	<b>87.45</b>	<b>83.12</b>	<b>84.10</b>
<i>RoBERTa Version</i>					
RoBERTa	87.40	93.19	89.17	86.50	89.07
WWM-RoBERTa	80.36	95.69	91.77	88.19	89.00
SimCSE-RoBERTa	76.65	96.20	88.58	89.49	87.73
SSCL-RoBERTa	80.71	<b>97.61</b>	92.50	<b>90.86</b>	90.42
W2CL-RoBERTa(Ours)	<b>90.20</b>	97.01	<b>94.81</b>	87.95	<b>92.49</b>

## 4 Analysis

### 4.1 Ablation Study

To evaluate the effectiveness of the two modules in the proposed method—the contrastive learning module and the word classification module, we conducted an ablation study. The experimental results are shown in Table 3. From the experimental results, it can be seen that removing the contrastive learning module leads to an average F1 score decrease of 0.37%, and removing the word classification module results in an average F1 score decrease of 0.34%. These findings indicate that the removal of either module results in a decline in the model’s performance.

### 4.2 Influence of Batch Size

During the training phase, the impact of batch size on the model’s final performance is illustrated in Table 4. As shown in the table, the model performs best when the batch size is set to 16. For the proposed W2CL method, a larger batch size not only increases the sample diversity within each batch during training but also provides more negative samples for contrastive learning. Additionally, some studies[24, 26] have demonstrated that the larger batch size enhances the effectiveness of contrastive learning.

**Table 3.** Ablation study for several methods evaluated on the sentence classification tasks. The evaluation metric is F1 score(%). CL: Contrastive Learning Module. WC: Word Classification Module.

Model	ASCC	CAIL	CHIP-CTC	FSCC	Avg.
W2CL-BERT	94.93	66.83	84.06	86.08	82.98
w/o CL	94.55(-0.38)	66.53(-0.30)	83.25(-0.81)	85.96(-0.12)	82.57(-0.41)
w/o WC	94.39(-0.54)	66.76(-0.07)	83.28(-0.78)	85.87(-0.21)	82.58(-0.40)
W2CL-RoBERTa	94.84	67.71	83.98	86.53	83.27
w/o CL	94.70(-0.14)	67.33(-0.38)	83.35(-0.63)	86.38(-0.15)	82.94(-0.33)
w/o WC	94.55(-0.29)	67.56(-0.15)	83.48(-0.50)	86.39(-0.14)	83.00(-0.27)

**Table 4.** The model’s performance under different batch size

Model	ASCC	CAIL(Wait)	CHIP-CTC	FSCC	Avg.
W2CL-BERT	94.93	66.83	84.06	86.08	82.98
bs=4	93.89(-1.04)	66.29(-0.54)	83.15(-0.91)	85.31(-0.77)	82.16(-0.82)
bs=8	94.42(-0.51)	66.45(-0.38)	83.42(-0.64)	85.83(-0.25)	82.53(-0.45)
W2CL-RoBERTa	94.84	67.71	83.98	86.53	83.27
bs=4	94.10(-0.74)	67.01(-0.70)	83.35(-0.63)	85.60(-0.93)	82.52(-0.75)
bs=8	94.55(-0.29)	67.26(-0.45)	83.56(-0.42)	86.13(-0.40)	82.88(-0.39)

## 5 Conclusion

This study proposes a novel framework called W2CL, which aims to better transfer general-purpose PLMs such as BERT and RoBERTa to specific domains, thereby improving the performance of the model in sentence classification tasks. First, we propose a domain knowledge extraction method based on ChatGPT. This method can extract high-quality domain knowledge from raw texts easily in different domains. Next, we introduce a multi-task learning approach that includes three tasks: WWM, contrastive learning, and word classification. This approach allows better integration of domain knowledge into the model. Our experimental results demonstrate that our proposed W2CL method outperforms other methods in sentence classification tasks across four domains, regardless of whether BERT or RoBERTa is used as the basic model. This also indicates that the W2CL method proposed in this study has general applicability.

## References

1. Chen, N., Shou, L., Gong, M., Pei, J., Cao, B., Chang, J., Jiang, D., Li, J.: Alleviating over-smoothing for unsupervised sentence representation. arXiv preprint arXiv:2305.06154 (2023)

2. Cui, Y., Che, W., Liu, T., Qin, B., Yang, Z.: Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **29**, 3504–3514 (2021). <https://doi.org/10.1109/TASLP.2021.3124365>
3. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. pp. 4171–4186 (2019)
4. Gao, T., Yao, X., Chen, D.: Simcse: Simple contrastive learning of sentence embeddings. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. pp. 6894–6910 (2021)
5. He, P., Liu, X., Gao, J., Chen, W.: Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654* (2020)
6. Jiang, T., Jiao, J., Huang, S., Zhang, Z., Wang, D., Zhuang, F., Wei, F., Huang, H., Deng, D., Zhang, Q.: Promptbert: Improving BERT sentence embeddings with prompts. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*. pp. 8826–8837 (2022)
7. Johnson, R., Zhang, T.: Supervised and semi-supervised text categorization using lstm for region embeddings. In: *International Conference on Machine Learning*. pp. 526–534. PMLR (2016)
8. Lauscher, A., Vulic, I., Ponti, E.M., Korhonen, A., Glavas, G.: Specializing unsupervised pretraining models for word-level semantic similarity. In: *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*. pp. 1371–1383 (2020)
9. Levine, Y., Lenz, B., Dagan, O., Ram, O., Padnos, D., Sharir, O., Shalev-Shwartz, S., Shashua, A., Shoham, Y.: Sensebert: Driving some sense into BERT. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. pp. 4656–4667 (2020)
10. Li, C., Gao, F., Bu, J., Xu, L., Chen, X., Gu, Y., Shao, Z., Zheng, Q., Zhang, N., Wang, Y., Yu, Z.: Sentiprompt: Sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis. *CoRR* (2021)
11. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019)
12. Liu, Z., Huang, Y., Yu, X., Zhang, L., Wu, Z., Cao, C., Dai, H., Zhao, L., Li, Y., Shu, P., et al.: Deid-gpt: Zero-shot medical text de-identification by gpt-4. *arXiv preprint arXiv:2303.11032* (2023)
13. Lyu, B., Chen, L., Zhu, S., Yu, K.: LET: linguistic knowledge enhanced graph transformer for chinese short text matching. In: *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. pp. 13498–13506 (2021)
14. Nov, O., Singh, N., Mann, D.M.: Putting chatgpt’s medical advice to the (turing) test. *medrxiv*. Preprint posted online January **24** (2023)
15. Peng, K., Ding, L., Zhong, Q., Shen, L., Liu, X., Zhang, M., Ouyang, Y., Tao, D.: Towards making the most of chatgpt for machine translation. *arXiv preprint arXiv:2303.13780* (2023)

16. Qin, C., Zhang, A., Zhang, Z., Chen, J., Yasunaga, M., Yang, D.: Is chatgpt a general-purpose natural language processing task solver? arXiv preprint arXiv:2302.06476 (2023)
17. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
18. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research* **21**(140), 1–67 (2020)
19. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108 (2019)
20. Shen, Y., Song, K., Tan, X., Li, D., Lu, W., Zhuang, Y.: Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems* **36** (2024)
21. Tai, K.S., Socher, R., Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. arXiv preprint arXiv:1503.00075 (2015)
22. Tian, H., Gao, C., Xiao, X., Liu, H., He, B., Wu, H., Wang, H., Wu, F.: SKEP: sentiment knowledge enhanced pre-training for sentiment analysis. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5–10, 2020*. pp. 4067–4076 (2020)
23. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017)
24. Wu, X., Gao, C., Zang, L., Han, J., Wang, Z., Hu, S.: Esimcse: Enhanced sample building method for contrastive learning of unsupervised sentence embedding. In: *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12–17, 2022*. pp. 3898–3907 (2022)
25. Xiao, C., Zhong, H., Guo, Z., Tu, C., Liu, Z., Sun, M., Feng, Y., Han, X., Hu, Z., Wang, H., et al.: Cail2018: A large-scale legal dataset for judgment prediction. arXiv preprint arXiv:1807.02478 (2018)
26. Xiao, S., Liu, Z., Zhang, P., Muennighof, N.: C-pack: Packaged resources to advance general chinese embedding. *CoRR* (2023)
27. Xu, Y., Zhu, C., Wang, S., Sun, S., Cheng, H., Liu, X., Gao, J., He, P., Zeng, M., Huang, X.: Human parity on commonsenseqa: Augmenting self-attention with external attention. In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23–29 July 2022*. pp. 2762–2768 (2022)
28. Zhang, N., Chen, M., Bi, Z., Liang, X., Li, L., Shang, X., Yin, K., Tan, C., Xu, J., Huang, F., et al.: Cblue: A chinese biomedical language understanding evaluation benchmark. arXiv preprint arXiv:2106.08087 (2021)
29. Zhu, X., Sobihani, P., Guo, H.: Long short-term memory over recursive structures. In: *International conference on machine learning*. pp. 1604–1612. PMLR (2015)