# A Survey of Pretrained Language Models

Kaili Sun[1], Xudong Luo[1(✉)], and Michael Y. Luo[2]

[1] Guangxi Key Lab of Multi-Source Information Mining and Security, School of Computer Science and Engineering, Guangxi Normal University, Guilin 541001, China
luoxd@mailbox.gxnu.edu.cn

[2] Emmanuel College, Cambridge University, Cambridge CB2 3AP, UK

**Abstract.** With the emergence of Pretrained Language Models (PLMs) and the success of large-scale PLMs such as BERT and GPT, the field of Natural Language Processing (NLP) has achieved tremendous development. Therefore, nowadays, PLMs have become an indispensable technique for solving problems in NLP. In this paper, we survey PLMs to help researchers quickly understand various PLMs and determine the appropriate ones for their specific NLP projects. Specifically, first, we brief on the main machine learning methods used by PLMs. Second, we explore early PLMs and discuss the main state-of-art PLMs. Third, we review several Chinese PLMs. Fourth, we compare the performance of some mainstream PLMs. Fifth, we outline the applications of PLMs. Finally, we give an outlook on the future development of PLMs.

**Keywords:** Machine learning · Natural Language Processing · Pretrained Language Models · BERT · GTP

## 1 Introduction

Pretrained Language Models (PLMs) are a new paradigm in Natural Language Processing (NLP) [22]. As shown in Fig. 1, a PLM is a large neural network. It is pre-trained on a large-scale text corpus through self-supervised learning (which is used to learn common sense from a large corpus with nothing to do with a specific downstream task). Pre-training can be regarded as regularisation to prevent the model from overfitting small data [16]. After being pre-trained, a PLM needs to be fine-tuned for a specific downstream task.

In early NLP tasks, low-dimensional and dense vectors are often used to represent language's syntactic or semantic features through various deep neural networks [32]. However, since deep neural networks usually have many parameters and the dataset used for training is limited, it may often lead to the phenomenon of overfitting. Transfer learning can apply the knowledge learnt in the source domain to the learning task in the target domain [40], alleviating the pressure caused by limited manual annotation data. However, unlabelled data is much larger than labelled data, so it is necessary to learn how to extract useful information from unlabelled data. The emergence of self-supervised learning


G. Memmi et al. (Eds.): KSEM 2022, LNAI 13369, pp. 442–456, 2022.
https://doi.org/10.1007/978-3-031-10986-7_36

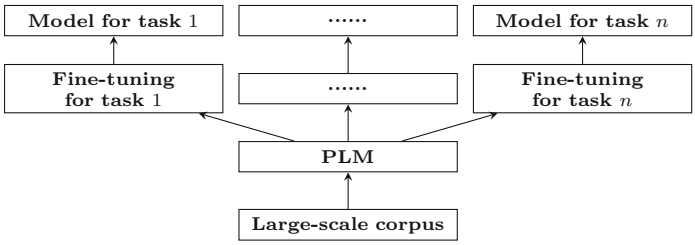


**Fig. 1.** The training process of a language model

and unsupervised learning solves this problem. Transformer [41] (a deep learning model) is proposed to solve the problem of slow training and low efficiency of Recurrent Neural Networks (RNNs) [36], and integrated with the self-attention mechanism to achieve fast parallel effects. Since then, PLMs have entered a boom phase. Large-scale PLMs such as BERT [11] and GPT [33] succeed greatly, and various improvements to them have been made to solve various NLP tasks.

Although PLMs are crucial to NLP tasks, there are not many surveys for helping researchers to quickly understand various PLMs from different viewpoints and determine the appropriate ones for their specific NLP projects. To amend this, in this paper, we provide a survey of PLMs. We found only two surveys on PLM through Google scholar, although ours in this paper is unique from them. The first one is provided by Li *et al.* [22], concerning the general task definition, the mainstream architectures of PLMs for text generation, the usage of existing PLMs to model different input data and satisfy unique properties in the generated text, and several critical fine-tuning strategies for text generation. However, they did not discuss the mainstream PLMs one by one as we do in this paper. The second survey we found was provided by Qiu *et al.* [32] in 2020. They comprehensively review PLMs, and, in particular, they systematically categorise various PLMs. However, the survey was published in March 2020, so it does not cover PLMs published afterwards, particularly Chinese PLMs in 2020 and 2021. So, instead, we cover the recent two years, especially the Chinese ones.

The rest of this paper is organised as follows. Section 2 briefs three main machine learning methods for training PLMs. Section 3 recalls early PLMs that focus on word vectors. Section 4 reviews the second generation of PLMs, including ELMo, BERT, GPT, and their derivatives. Section 5 briefs several Chinese PLMs and compares them with several typical English PLMs. Section 6 lists the main NLP tasks for which PLMs can be used and gives an application example for each task. Finally, Sect. 7 summarises this paper with the future work.

## 2 Basic Machine Learning Methods for PLMs

This section will brief machine learning methods for PLMs: Long-Short Term Memory (LSTM) [19], Attention Mechanism (AM) [6], and Transformer [41].

### 2.1 Long-Short Term Memory

RNNs are often used to process sequence data such as machine translation and sentiment analysis, but they are short-term memory networks. When faced with a long enough data sequence, it is difficult to transmit the earlier information to them later because RNNs may meet gradient disappearance in the reverse transmission. LSTM is an improved RNN model. Based on RNN, an input gate, a forgetting gate and an output gate are added to control and retain information, which overcomes the limitation of short-term memory. The forget gate controls how much of the unit status at the last moment can be retained to the current moment. The input gate determines how much of the immediate status can be input into the unit status. Finally, the output gate is responsible for controlling how much of the unit status can be used as the current output value of the LSTM. However, in both RNN and LSTM, much of the information carried by the input word vector may be lost in the face of long sequences.

### 2.2 Attention Mechanism

AM sets a weight for all hidden states in the encoder and inputs the information of the hidden states after the weighted summation to the decoder layer. AM pays more attention to inputs relevant to the current task. The AM acts between the encoder and the decoder. When RNNs are integrated with the attention mechanism, they can predict a particular part of the output sequence and focus their attention on a specific part of the input sequence to generate a higher quality output. Thus, Yu *et al.* [49] integrated LSTM with an AM and two-way LSTM for the Chinese question answering system, which solves the difficulties caused by Chinese grammar, semantics and lexical limitations in the Chinese question answering dataset. The AM+LSTM model retains the intermediate outputs of the LSTM encoder on the input sequences and then trains the model to selectively learn these inputs and associate the output sequences with the model outputs.

Later on, a self-AM was proposed [41]. The self-AM acts on the encoder or the decoder, and can connect longer-distance words in the same sentence. General embedding methods, such as Word2Vec, need to be integrated with context to clarify the semantics, and the sequence information of the sentence is lost. Self-AM can effectively solve these problems. Moreover, self-AM replaces the most commonly used loop layer in the encoder-decoder architecture with multi-headed self-attention. Multi-headed attention focuses on information from different representation subspaces in different positions, leading to a dramatic improvement in training speed [24,27].

### 2.3 Transformer

Transformer [41] uses multiple encoders and decoders. The encoder contains a self-attention layer and a feed-forward neural network in addition to the self-attention layer and the feed-forward neural network. The advantage of the Transformer model is that it can solve the problems of slow training and low efficiency

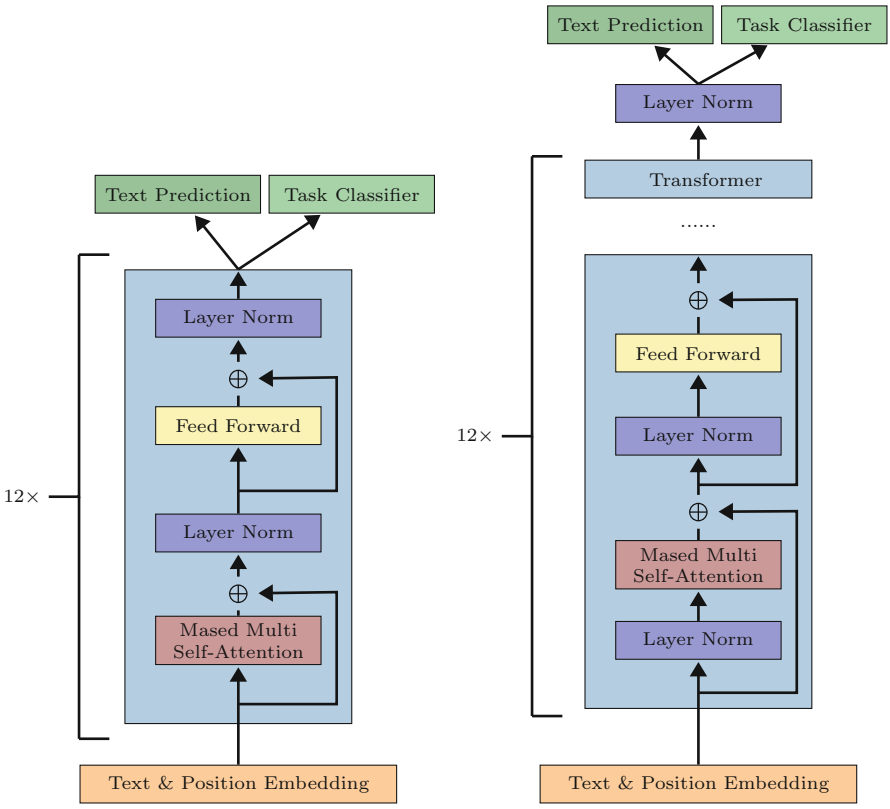


**Fig. 2.** Transformer architecture used by GPT and GPT-2

of the RNN model and use self-attention to achieve fast parallel effects. Moreover, it can deeply mine the characteristics of a Deep Neural Network (DNN) to improve the efficiency and performance of the model. After the Transformer model was proposed, PLMs entered a boom phase. Figure 2 shows the Transformer architectures used by GPT [33] and GPT-2 [34].

## 3 Early PLMs

From 2013 to 2021, PLMs have upgraded year by year. As early as 2013, Mikolov *et al.* [28] proposed the first PLM, called Word2Vec, which generates word vectors. According to the corpus, the optimised model trained expresses a word as a vector quickly and effectively. After Word2Vec trains the word vector, each independent word has a fixed dimension vector corresponding to its semantics. The Word2Vec model is also a widely used word embedding model in sentiment analysis with excellent analysis performance [1]. The Word2Vec model uses two

algorithms for generating word vectors, Skip-Gram (SG) and Continuous Bag of Words (CBoW). SG has a good performance on small training data, but CBoW is also very efficient in big training data, and its accuracy rate for frequent words is also higher than SG.

Later on, Pennington, Socher, and Manning [30] proposed GloVe to overcome the shortcomings of Word2Vec: its vector dimensionality is low, and it cannot completely cover the data in the corpus. Moreover, GloVe can be more generalised than Word2Vec in the process of word embedding. However, GloVe uses the matrix factorisation method and the method based on shallow windows. So, it can contain local or global information of specific words in the corpus, which is necessary for improving its performance.

Both Word2Vec and GloVe map the input to a fixed-dimensional vector representation, so the generated word vectors are all context-independent. Thus, they cannot handle linguistic phenomena like polysemous words. For example, "open" has different meanings in "the door is open" and "the farm is in the open countryside". So, it is unreasonable that its word vectors for the two sentences are the same. Moreover, these models are no longer needed in downstream tasks because their computational efficiency usually is low.

## 4 Second Generation of PLMs

This section will review representative PLMs of second generation.

### 4.1 ELMo Model

To solve the problem of polysemy and understand complex context, in 2018, Peters *et al.* [31] proposed ELMo (Embedding from Language Models). It learns word vectors via the internal state of a deep bidirectional language model. It extracts embeddings from a bi-directional LSTM pre-trained on a sizeable unsupervised corpus. The resulting embeddings are derived from a weighted combination of internal layers that can be easily applied to existing models. When doing the downstream task, ELMo extracts word embeddings from a pre-trained network corresponding to words from each layer of the network as new embeddings to be added to the downstream task. It is a typical PLM residing in feature fusion. In many NLP tasks in different domains, ELMo performs very well [3,15,47].

### 4.2 BERT Family

ELMo is a one-way language model, and its ability to model semantic information is limited. To remove these limitations, Google AI launched pre-training language model BERT (Bidirectional Encoder Representations from Transformers) at the end of 2018 [11], which uses Masked Language Model (MLM) and Next Sentence Prediction (NSP) for deep two-way joint training. The task of

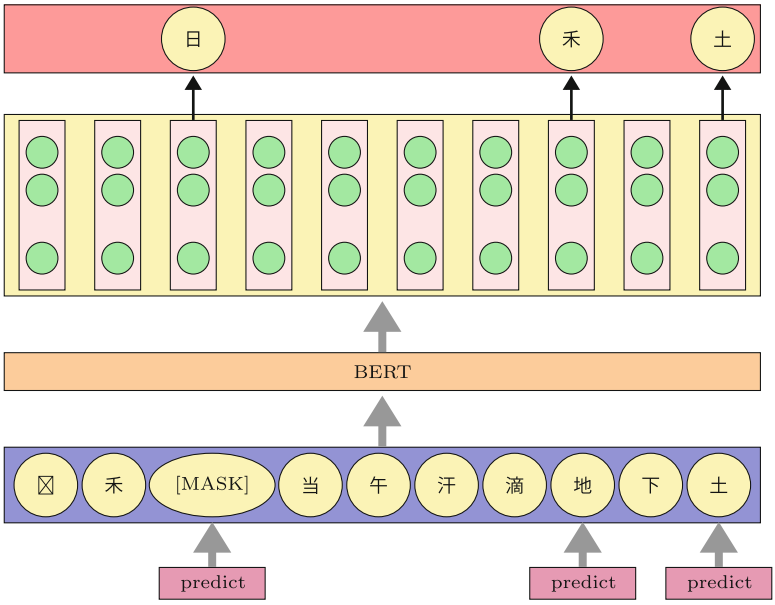


**Fig. 3.** Masked LM

MLM is to randomly erase one or several words in a given sentence and predict the erased words according to the remaining words. For the erased words, MASK can replace 80% of cases; for any word, it can replace 10% of cases; and for unchanged words, it can do 10% of cases. This is more conducive to the model to grasp the context information. Figure 3 shows the MLM process. Moreover, BERT uses NSP to capture the relationship between sentences [11]. Its performance, ease of use and versatility surpass many models. The difference between BERT's pre-training and downstream specific task training is only the top-level output layer, and it can be used in many tasks. BERT has achieved significant improvements in 11 basic tasks in NLP.

The emergence of BERT has extensively promoted the development of the NLP field. Since its emergence, researchers have proposed many improved models based on BERT. RoBERTa [23] uses a more extensive dataset, changes the static mask to the dynamic mask, and cancels the NSP task. AlBERT [21] can share parameters cross-layer, which significantly reduces parameters. It factors Embedding into two smaller embedding matrices and changes the NSP task in BERT to SOP (sentence-order prediction). XLNet [48] uses Permutation Language Modeling (PLM), which can capture contextual information in the language model and has apparent advantages over BERT in text generating tasks with long document input. ELECTRA [7] replaces the MLM in BERT with RTD (Replaced Token Detection), which solves the inconsistency between the pre-training phase and the fine-tuning phase of MASK. ELECTRA is better than BERT under the same computing power, data, and model parameters. It is also better than RoBERTa and XLNet under the same amount of calculation.

### 4.3 GPT Family

Although the unlabelled text corpus is rich, there is very little labelled data for learning for specific tasks. To address the issue, in 2018, Radford *et al.* [33] proposed the GPT (Generative Pre-trained Transformer) model. Its pre-training includes two stages. The first stage is unsupervised pre-training, learning high-capacity language models on a large number of text corpora. The unsupervised pre-training in GPT mainly uses the decoder layer in the transformer; Fig. 2(a) shows the process. First, the sum of the word vector and the position vector is input. Then, after 12 layers of transformers, the predicted vector and the vector of the last word are obtained. Finally, the word vector of the last word will be used as the input of subsequent fine-tuning. The second stage is supervised fine-tuning, which adapts the model to discriminative tasks with labelled data. GPT is the first model that integrates the modern Transformer architecture and the self-supervised pre-training objective [45]. It surpasses the previous model in various assessments of natural language reasoning, classification, question answering, and comparative similarity.

In 2019, Radford *et al.* [34] proposed GPT-2 to predict the next word in a sentence. GTP-2 can answer questions, summarise texts, and translate texts without training in a specific field. However, GPT-2 still uses the one-way transformer mode of GPT with simple adjustments. Figure 2(b) shows the minimum model of GPT-2. It puts layer normalisation before each sub-block and adds a layer normalisation after the last self-attention. The training data of GPT-2 has been greatly improved in quantity, quality, and breadth. However, the network parameters have also increased, and the network parameters of the largest GPT-2 has reached 48 layers. As a result, both Zero-Shot (especially the tiny dataset Zero-Shot) and long text (long-distance dependence) perform well.

In 2020, Brown *et al.* [5] proposed GPT-3, which has 175 billion parameters. GPT-3 has excellent performance on many NLP datasets, including translation, question answering, and text filling. It is highly efficient, especially in text generation, and it is almost indistinguishable from a human-generated text. Although GPT-3 has made significant progress, it does not follow the real intentions of users very well, and it often produces unreal, harmful or unresponsive emotional outputs. To remove this flaw, Open AI uses reinforcement learning from human feedback to fine-tune GPT-3 - the resulting fine-tuned model is called InstructGPT [29]. The three main steps of its training process are: 1) perform supervised learning with a manually written demo dataset, 2) train the reward model RM on this dataset, and 3) use RM as the reward function for reinforcement learning. After over a year of testing, the experiments show that although InstructGPT still has simple errors, compared with GPT-3, it reduces harmful output and significantly improves its ability to follow user intentions.

## 5 Chinese PLM

This section will discuss important Chinese PLMs.

### 5.1 PLMs from IFLYTEK and Harbin Institute of Technology

In 2019, Cui *et al.* [9] of IFLYTEK and Harbin Institute of Technology released the Chinese PLM BERT-wwm based on the whole word mask. For Chinese, if part of a complete word is masked, other parts of the same word also are masked. Their experiments show that BERT-wwm outperforms BERT in various Chinese NLP tasks. They also increase the training data level and the training steps to upgrade BERT-wwm to BERT-wwm-ext. In addition, they also proposed a series of Chinese PLMs based on BERT, such as RoBERTa [26], in the same year, which achieved good experimental results. In 2020, Cui *et al.* [8] trained the Chinese PLM XLNet-mid based on the XLNet open source code using large-scale Chinese corpus. As a result, it surpassed the effects of BERT-wwm and BERT-wwm-ext on most NLP tasks and achieved significant performance improvements in machine reading comprehension tasks.

### 5.2 ERNIE Family from Baidu

In 2019, Zhang *et al.* [51] in Baidu released the Chinese PLM ERNIE, which has a greatly enhanced general semantic representation ability by uniformly modelling the grammatical structure, lexical structure, and semantic information in its training data. Moreover, they use a higher quality Chinese corpus, making ERNIE more effective on Chinese NLP tasks. Their experiments show that on 5 Chinese NLP tasks, ERNIE surpassed BERT. In December 2019, ERNIE topped the list in the authoritative dataset GLUE (General Language Understanding Evaluation) in the field of NLP.[1] In 2020, Sun *et al.* [39] released ERNIE 2.0. This model extracts more valuable information from the training corpus through continuous multi-task learning. Their experiments show that ERNIE 2.0 outperforms BERT and XLNet on 16 tasks, including the English task on the GLUE benchmark and several similar tasks in Chinese. In 2021, Sun *et al.* [38] released ERNIE 3.0 by integrating autoregressive and autoencoder networks with general semantic layers and task-related layers. Once the pre-training of the generic semantic layer is completed, it is not updated anymore. Only task-dependent layers are fine-tuned when performing downstream tasks, significantly improving efficiency. Their experiments show that the model outperforms state-of-the-art models on 54 Chinese NLP tasks.

### 5.3 TinyBERT from Huawei

Large-scale PLMs such as BERT have huge parameters and complex computing processes, making it challenging to apply them on edge devices with limited computing power and memory. To this end, many model compression techniques have been proposed, mainly including quantisation [17], weights pruning [18], and knowledge distillation [35]. In 2019, Huawei proposed TinyBERT [20], which uses a new knowledge distillation method to perform transformer distillation

[1] https://gluebenchmark.com/.

**Table 1.** Comparison of the characteristics of some pre-trained models

| PLMs | Characteristic | | | |
|---|---|---|---|---|
| | Learning method | Language model | Language type | Params |
| Elmo [7] | LSTM | BiLM | English | 96M |
| GPT [33] | Transformer Dec | LM | English | 117M |
| BERT [11] | Transformer Enc | MLM | English | 110M |
| RoBERTa [26] | Transformer Enc | MLM+ RTD | English | 355M |
| ELECTRA [7] | Transformer Enc | MLM | English | 335M |
| BERT-wwm-ext [9] | Transformer Enc | PLM | Chinese | 108M |
| XLNet-mid [46] | Transformer Enc | MLM+ DEA | Chinese | 209M |
| ERNIE [51] | Transformer Enc | 95.06 | Chinese | 114M |

in pre-training and task-specific learning stages. This two-stage learning framework enables TinyBERT to acquire a general knowledge of "teacher" BERT and task-specific knowledge. The research results show that although the size of TinyBERT is only 13.3% of BERT, its computing speed is 9.4 times that of BERT, and the testing effect on the GLUE benchmark is comparable to BERT.

### 5.4 WuDao Family from BAAI

In March 2021, the Beijing Academy of Artificial Intelligence (BAAI) released large-scale Chinese PLM WuDao 1.0, called WenHui.[2] It replaces the Transformer model in GPT with Transformer-XL [10], generating human-based text and better maintaining content consistency. It can also learn concepts between different modalities, overcoming the limitation of large-scale self-supervised PLMs that do not possess such cognitive capabilities. Two months later, BAAI released WuDao 2.0 with a parameter volume of 1.75 trillion.[3] China's first trillion-level PLM with ten times the number of parameters than GPT-3. Wudao 2.0 can be applied not only to a single text field but also to the visual field. It can generate pictures according to text, and it can also retrieve text according to pictures. WuDao 2.0 achieved first place in 9 benchmarks in terms of precision.[4]

### 5.5 PLUG from Alibaba Dharma Academy

In April 2021, Alibaba Dharma Academy released the world's largest Chinese text PLM, PLUG (Pre-training for Language Understanding and Generation).[5] According to the strengths of their NLU (Natural Language Understanding)

[2] https://mp.weixin.qq.com/s/BUQWZ5EdR19i40GuFofpBg.
[3] https://mp.weixin.qq.com/s/NJYINRt_uoKAIgxjNyu4Bw.
[4] https://wudaoai.cn/home.
[5] https://m.thepaper.cn/baijiahao_12274410.

**Table 2.** Performance comparison of some pretrained models

| PLMs | Dataset | | | Results | | |
|---|---|---|---|---|---|---|
| | GLUE dev set | SST-2 | CMRC | F1-score | Accuracy | GLUE |
| Elmo [7] | ✓ | | | N/A | N/A | 71.2 |
| GPT [33] | | ✓ | | N/A | 91.3 | 72.8 |
| BERT [11] | | ✓ | | N/A | 94.9 | 82.1 |
| RoBERTa [26] | | ✓ | | N/A | 96.7 | 88.1 |
| ELECTRA [7] | | ✓ | | N/A | 97.1 | 89.4 |
| BERT-wwm-ext [9] | | | ✓ | 73.23 | N/A | N/A |
| XLNet-mid [46] | | | ✓ | 66.51 | N/A | N/A |
| ERNIE [51] | | ✓ | | N/A | 97.8 | 91.1 |

language model StructBERT [44] and NLG (Natural Language Generation) language model PALM [33], they jointly train NLU & NLG of PLUD. The joint training makes PLUG understand an input text better and generate more relevant content accordingly. It also uses more than 1TB of high-quality Chinese training datasets, setting a new record for Chinese GLUE with a score of 80.614 on language understanding tasks. PLUD performs excellently in long text generation such as novel creation, poetry generation, and intelligent question answering. Its goal is to surpass humans in various tasks of Chinese NLP.

### 5.6 Comparison of Some Chinese and English PLMs

Table 1 and Table 2 compares some Chinese and English PLMs on datasets GLUE/CLUE, MRPC, and SST-2. GLUE is a benchmark dataset for evaluating and analysing the performances of various models in various existing NLU tasks [42], and CLUE is a Chinese NLU evaluation benchmark [46]. The SST-2 (Stanford Sentiment Treebank v2) dataset consists of 215,154 phrases with fine-grained sentiment labels from movie reviews [37]. MRPC (Microsoft Research Paraphrase Corpus), introduced by Dolan *et al.* [13], is a corpus of 5,801 sentence pairs collected from newswire articles.

## 6 Practical Applications of Pretrained Models

This section will briefly review the main NLP tasks that PLMs can be applied.

### 6.1 Sentiment Analysis

During the COVID-19 pandemic, it is critical to identify negative public sentiment characteristics and adopt scientific guidance to alleviate the public's concerns. To more accurately analyse the sentiment of online reviews, Wang *et al.* [43] first uses unsupervised BERT to classify the sentiment of the collected text

and then uses the TF-IDF algorithm to extract text topics. The accuracy of this method outperforms all baseline NLP algorithms.

### 6.2 Named Entity Recognition

To solve the accuracy of biomedical nomenclature recognition in low-resource languages and improve the efficiency of text reading, Boudjellal *et al.* [4] proposed a model named ABioNER based on BERT. They first pre-trained AraBERT on a general-domain Arabic corpus and a corpus of biomedical Arabic literature and then fine-tuned AraBERT using a single NVIDIA GPU. Their test result values demonstrate that building a monolingual BERT model on small-scale biomedical data can improve understanding of data in the biomedical domain.

### 6.3 Summarisation

Liu, Wu, and Luo [25] proposed a method for summarising legal case documents. First, they extract five key components of a legal case document. Thus, the text summarisation problem becomes five text compression and integration problems for sentences of five different categories. Then they fine-tune five models of PLM GPT-2 for each key component. Next, they use the five fine-tuned models to conduct text compression and integration for summarising each key component. Finally, they put all the summaries of five key components together to obtain the summary of the entire legal case document. They did lots of experiments to confirm the effectiveness of their approach.

### 6.4 Question Answering

The current BERT-based question answering systems suffer several problems. For example, a system of this kind may return wrong answers or nothing, cannot aggregate questions, and only consider text contents but ignore the relationship between entities in the corpus. As a result, the system may not be able to validate its answer to a question. To address these issues, Do and Phan [12] developed a question answering system based BERT and knowledge graph. They used BERT to build two classifiers: (1) BERT-based text classification for content information and (2) BERT-based triple classification for link information. Their experiments show that their method significantly outperformed the state-of-the-art methods in terms of accuracy and executive time.

### 6.5 Machine Translation

Zhang *et al.* [50] proposed a BERT-based method for machine translation, called BERT-JAM. The proposed method has the following features. First, BERT-JAM fuses BERT's multi-layer representations into an overall representation that the neural machine translation model can use. Second, BERT-JAM can dynamically integrate the BERT representation with the encoder/decoder representations.

Third, they fine-tune BERT-JAM using a three-phase optimisation strategy. The strategy can gradually ablate different components to beat catastrophic forgetting during fine-tuning. Their experiments show that the performance of BERT-JAM on multiple translation tasks is state-of-the-art.

## 7 Conclusions

Before being fine-tuned, PLMs already perform very well. After fine-tuning, their performances are even better, and the fine-tuned models are well-converged. Therefore, PLMs have been used for many NLP tasks [2,11,14,48]. Thus, this paper provides a survey on PLMs to help researchers quickly understand various PLMs and determine which ones are appropriate for their specific NLP projects. Specifically, we brief the main machine learning methods used by PLMs and review early PLMs, main state-of-art PLMs, and several well-known Chinese PLMs. Moreover, we compare the performance of some mainstream PLMs. In addition, we list the main NLP tasks for which PLMs have been used and review some state-of-art work for each task of them.

Although the emergence and application of PLMs have promoted the rapid development of many NLP tasks, due to the complexity of natural language, PLM technology still faces many challenges. First of all, the performance of PLMs is far from reaching its upper limit. Longer training steps and larger datasets could potentially improve its performance. Secondly, fine-tuning is required when applying PLM to downstream tasks, but the fine-tuning is specific, which may result in low efficiency. When applying PLMs in specialised fields such as biomedical science and law, PLMs may be susceptible to learning and amplifying biases in datasets due to the specificity of datasets in specialised fields. For example, a PLM may generate biases against age groups and gender. Finally, there are many different languages, and many ways to express their linguistic information. So, a single pre-trained language model cannot meet people's needs fully. Hence, multi-lingual PLMs and multi-modal PLMs have become a particular focus of attention as it is vital to improve their performance to meet various needs now and in the future.

**Acknowledgment.** This work was supported by the National Natural Science Foundation of China (No. 61762016) and the Graduate Student Innovation Project of School of Computer Science and Engineering, Guangxi Normal University (JXXYYJSCXXM-2021-001).

## References

1. Alnawas, A., Arici, N.: Effect of word embedding variable parameters on Arabic sentiment analysis performance. arXiv preprint arXiv:2101.02906 (2021)
2. Bao, H., et al.: UniLMv2: pseudo-masked language models for unified language model pre-training. In: Proceedings of the 37th International Conference on Machine Learning, pp. 642–652 (2020)

3. Barlas, G., Stamatatos, E.: Cross-domain authorship attribution using pre-trained language models. In: Maglogiannis, I., Iliadis, L., Pimenidis, E. (eds.) AIAI 2020. IAICT, vol. 583, pp. 255–266. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49161-1_22
4. Boudjellal, N., et al.: ABioNER: a BERT-based model for Arabic biomedical named-entity recognition. Complexity **2021**, 1–6 (2021)
5. Brown, T., et al.: Language models are few-shot learners. In: Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901 (2020)
6. Chaudhari, S., Mithal, V., Polatkan, G., Ramanath, R.: An attentive survey of attention models. ACM Trans. Intell. Syst. Technol. **12**(5), 1–32 (2021)
7. Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: ELECTRA: pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555 (2020)
8. Cui, Y., Che, W., Liu, T., Qin, B., Wang, S., Hu, G.: Revisiting pre-trained models for Chinese natural language processing. In: Findings of the Association for Computational Linguistics, EMNLP 2020, pp. 657–668 (2020)
9. Cui, Y., Che, W., Liu, T., Qin, B., Yang, Z.: Pre-training with whole word masking for Chinese BERT. IEEE/ACM Trans. Audio Speech Lang. Process. **29**, 3504–3514 (2021)
10. Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q.V., Salakhutdinov, R.: Transformer-XL: attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860 (2019)
11. Devlin, J., Chang, M.W., 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, pp. 4171–4186 (2019)
12. Do, P., Phan, T.H.V.: Developing a BERT based triple classification model using knowledge graph embedding for question answering system. Appl. Intell. **52**(1), 636–651 (2021). https://doi.org/10.1007/s10489-021-02460-w
13. Dolan, B., Brockett, C.: Automatically constructing a corpus of sentential paraphrases. In: Proceedings of the 3rd International Workshop on Paraphrasing, pp. 9–16 (2005)
14. Dong, L., et al.: Unified language model pre-training for natural language understanding and generation. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems, pp. 13063–13075 (2019)
15. El Boukkouri, H., Ferret, O., Lavergne, T., Noji, H., Zweigenbaum, P., Tsujii, J.: CharacterBERT: reconciling ELMo and BERT for word-level open-vocabulary representations from characters. In: Proceedings of the 18th International Conference on Computational Linguistics, pp. 6903–6915 (2020)
16. Erhan, D., Courville, A., Bengio, Y., Vincent, P.: Why does unsupervised pre-training help deep learning? In: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, pp. 201–208 (2010)
17. Gong, Y., Liu, L., Yang, M., Bourdev, L.: Compressing deep convolutional networks using vector quantization. arXiv preprint arXiv:1412.6115 (2014)
18. Han, S., Pool, J., Tran, J., Dally, W.: Learning both weights and connections for efficient neural network. In: Advances in Neural Information Processing Systems 28 (2015)
19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
20. Jiao, X., et al.: TinyBERT: distilling BERT for natural language understanding. arXiv preprint arXiv:1909.10351 (2019)

21. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: a lite BERT for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942 (2019)
22. Li, J., Tang, T., Zhao, W., Wen, J.: Pretrained language models for text generation: a survey. In: Proceedings of the 30th International Joint Conference on Artificial Intelligence, pp. 4492–4497 (2021)
23. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: VisualBERT: a simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557 (2019)
24. Lin, Y., Wang, C., Song, H., Li, Y.: Multi-head self-attention transformation networks for aspect-based sentiment analysis. IEEE Access **9**, 8762–8770 (2021)
25. Liu, J., Wu, J., Luo, X.: Chinese judicial summarising based on short sentence extraction and GPT-2. In: Qiu, H., Zhang, C., Fei, Z., Qiu, M., Kung, S.-Y. (eds.) KSEM 2021. LNCS (LNAI), vol. 12816, pp. 376–393. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-82147-0_31
26. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
27. Meng, Z., Tian, S., Yu, L., Lv, Y.: Joint extraction of entities and relations based on character graph convolutional network and multi-head self-attention mechanism. J. Exp. Theoret. Artif. Intell. **33**(2), 349–362 (2021)
28. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
29. Ouyang, L., et al.: Training language models to follow instructions with human feedback. arXiv preprint arXiv:2203.02155 (2022)
30. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)
31. Peters, M., et al.: Deep contextualized word representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 2227–2237 (2018)
32. Qiu, X.P., Sun, T.X., Xu, Y.G., Shao, Y.F., Dai, N., Huang, X.J.: Pre-trained models for natural language processing: a survey. Sci. Chin. Technol. Sci. **63**(10), 1872–1897 (2020). https://doi.org/10.1007/s11431-020-1647-3
33. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018). https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf
34. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI Blog **1**(8), 9 (2019)
35. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: FitNets: hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
36. Sherstinsky, A.: Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D **404**, 132306 (2020)
37. Socher, R., et al.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631–1642 (2013)
38. Sun, Y., et al.: ERNIE 3.0: large-scale knowledge enhanced pre-training for language understanding and generation. arXiv preprint arXiv:2107.02137 (2021)
39. Sun, Y., et al.: ERNIE 2.0: a continual pre-training framework for language understanding. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 8968–8975 (2020)

40. Torrey, L., Shavlik, J.: Transfer learning. In: Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques, pp. 242–264. IGI Global (2010)
41. Vaswani, A., et al.: Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 6000–6010 (2017)
42. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: GLUE: a multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461 (2018)
43. Wang, T., Lu, K., Chow, K.P., Zhu, Q.: COVID-19 sensing: negative sentiment analysis on social media in China via BERT model. IEEE Access **8**, 138162–138169 (2020)
44. Wang, W., et al.: StructBERT: incorporating language structures into pre-training for deep language understanding. arXiv preprint arXiv:1908.04577 (2019)
45. Xu, H., et al.: Pre-trained models: past, present and future. arXiv preprint arXiv:2106.07139 (2021)
46. Xu, L., et al.: CLUE: a Chinese language understanding evaluation benchmark. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 4762–4772 (2020)
47. Yang, M., Xu, J., Luo, K., Zhang, Y.: Sentiment analysis of Chinese text based on Elmo-RNN model. J. Phys: Conf. Ser. **1748**(2), 022033 (2021)
48. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: XLNet: generalized autoregressive pretraining for language understanding. Adv. Neural. Inf. Process. Syst. **32**, 5753–5763 (2019)
49. Yu, X., Feng, W., Wang, H., Chu, Q., Chen, Q.: An attention mechanism and multi-granularity-based Bi-LSTM model for Chinese Q&A system. Soft. Comput. **24**(8), 5831–5845 (2019). https://doi.org/10.1007/s00500-019-04367-8
50. Zhang, Z., Wu, S., Jiang, D., Chen, G.: BERT-JAM: maximizing the utilization of BERT for neural machine translation. Neurocomputing **460**, 84–94 (2021)
51. Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., Liu, Q.: ERNIE: enhanced language representation with informative entities. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1441–1451 (2019)