electronics

Article

Research on the Application of Prompt Learning Pretrained Language Model in Machine Translation Task with Reinforcement Learning

Canjun Wang [1,2], Zhao Li [1,2,3,*], Tong Chen [4], Ruishuang Wang [4] and Zhengyu Ju [1,2]

[1] Faculty of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China; 10431210713@stu.qlu.edu.cn (C.W.); 10431210623@stu.qlu.edu.cn (Z.J.)
[2] Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan 250013, China
[3] Shandong Key Laboratory of Computer Networks, Jinan 250000, China
[4] Innovation Research Institute, Shandong Yiyun Information Technology Co., Ltd., Jinan 250101, China; chentong@sdas.org (T.C.); wangrs@sdas.org (R.W.)
* Correspondence: liz@sdas.org

Abstract: With the continuous advancement of deep learning technology, pretrained language models have emerged as crucial tools for natural language processing tasks. However, optimization of pretrained language models is essential for specific tasks such as machine translation. This paper presents a novel approach that integrates reinforcement learning with prompt learning to enhance the performance of pretrained language models in machine translation tasks. In our methodology, a “prompt” string is incorporated into the input of the pretrained language model, to guide the generation of an output that aligns closely with the target translation. Reinforcement learning is employed to train the model in producing optimal translation results. During this training process, the target translation is utilized as a reward signal to incentivize the model to generate an output that aligns more closely with the desired translation. Experimental results validated the effectiveness of the proposed approach. The pretrained language model trained with prompt learning and reinforcement learning exhibited superior performance compared to traditional pretrained language models in machine translation tasks. Furthermore, we observed that different prompt strategies significantly impacted the model’s performance, underscoring the importance of selecting an optimal prompt strategy tailored to the specific task. The results suggest that using techniques such as prompt learning and reinforcement learning can improve the performance of pretrained language models for tasks such as text generation and machine translation. The method proposed in this paper not only offers a fresh perspective on leveraging pretrained language models in machine translation and other related tasks but also serves as a valuable reference for further research in this domain. By combining reinforcement learning with prompt learning, researchers can explore new avenues for optimizing pretrained language models and improving their efficacy in various natural language processing tasks.

Keywords: reinforcement learning; prompt learning; pretrained language model; machine translation; natural language processing

check for updates

Citation: Wang, C.; Li, Z.; Chen, T.; Wang, R.; Ju, Z. Research on the Application of Prompt Learning Pretrained Language Model in Machine Translation Task with Reinforcement Learning. *Electronics* **2023**, *12*, 3391. https://doi.org/10.3390/electronics12163391

Academic Editor: Arkaitz Zubiaga

Received: 7 July 2023
Revised: 5 August 2023
Accepted: 6 August 2023
Published: 9 August 2023

1. Introduction

With the rapid advancement of technology, there is a growing interest in natural language processing (NLP), a field that encompasses various technologies such as speech recognition, automatic text translation, and intelligent customer service [1]. Among these technologies, machine translation has emerged as a crucial tool for cross-lingual communication and information dissemination, attracting significant attention and widespread usage [2]. Machine translation aims to convert text from one natural language to another and involves interdisciplinary fields such as linguistics, computer science, and statistics [3].

Traditional machine translation methods include rule-based, statistical, and deep learning approaches [4]. Rule-based methods rely on manually crafted rules for translation, providing precise control over the translation process. However, they are time-consuming and struggle with the flexibility and ambiguity of language [5]. Statistical methods, on the other hand, leverage large parallel corpora for modeling and learning, allowing them to handle the flexibility and ambiguity in language. However, they often require substantial data and computational resources, and may not yield high-quality translations of low-resource languages or specialized domains [6]. In recent years, deep learning methods, particularly those based on neural networks, have gained prominence in machine translation. These approaches learn universal language representations by training on extensive parallel corpora, enabling end-to-end translation [7]. Neural networks offer powerful expressive capabilities for handling nonlinear relationships and high-dimensional features, leading to an improved translation performance [8]. Nonetheless, deep learning methods heavily rely on annotated data for training, which can be challenging to obtain, especially for low-resource languages and specialized domains. This limitation has spurred the development of unsupervised pretraining language model (PLM) methods [9]. Unsupervised PLMs leverage large amounts of unlabeled corpora to learn universal language representations, thereby enhancing performance on specific tasks. Prominent models such as BERT (bidirectional encoder representations from transformers) and GPT (generative pretrained transformer) have emerged as advanced NLP technologies [10]. These models have revolutionized the field by capturing contextual information and achieving state-of-the-art results in various NLP tasks. The field of machine translation has witnessed significant advancements, with deep learning methods, particularly neural networks, gaining prominence. However, the dependence on annotated data has hindered their application in low-resource languages and specialized domains. To address this challenge, unsupervised PLM methods have emerged as a promising solution. Models such as BERT and GPT have been pioneering technologies, pushing the boundaries of natural language processing.

A pretrained language model (PLM) is an emerging NLP technology that can learn language representations from a large amount of text data, thus achieving significant results in various NLP tasks. The emergence of PLMs has alleviated the problem of data scarcity, which has been a long-standing challenge in the field of NLP. The advantage of PLMs is that they can learn language representations from a large amount of text data in an unsupervised manner, and these representations can be applied to various NLP tasks, such as text classification, sentiment analysis, named entity recognition, question-answering systems, and machine translation. For example, in machine translation tasks, PLMs may not understand the two languages well, leading to errors in translation. In addition, the large amount of computing resources required for PLMs may limit their application in low-resource environments. In recent years, researchers have proposed a new approach to address this issue, which is to combine reinforcement learning with prompt learning [11]. Reinforcement learning (RL) is a machine learning method that trains an agent to take actions in a specific environment, to achieve a particular goal through trial-and-error and reward-punishment mechanisms. RL typically consists of three components: the environment, agent, and policy. The environment refers to the specific environment in which the agent operates, the agent is an algorithm that achieves a specific goal through learning and training, and the policy refers to the way the agent takes action in a specific environment. In NER, reinforcement learning can be used to optimize the label propagation algorithm to improve recognition accuracy [12]. Prompt learning is a technique that can be combined with reinforcement learning. Its basic idea is to help models fine-tune specific tasks by introducing some prompt information. For example, in machine translation tasks, specific words or phrases about language translation can be provided to the model to help it better understand the differences between languages. Prompt learning can be used in many NLP tasks, such as text classification, sentiment analysis, and named entity recognition. Its advantage is that it can improve the performance of the model by providing targeted prompt information [13].

In the combined approach of reinforcement learning and prompt learning, the model can learn to select the best action given the prompt information through reinforcement learning [14]. In the field of pretrained language models, prompt learning has been extensively studied and applied to various natural language processing tasks [15] Prompt learning guides the model to learn specific language tasks by adding some prompt information to the model's input [16–20]. In GPT-3, prompt sentences are used to guide the model to generate text consistent with the prompt. Additionally, GPT-3 uses task-specific prompts, such as including a question and context in the input for question answering tasks and using appropriate prompts to indicate the generated answer [9,21–24]. Similarly, the T5 model adds prompts as tokens to the input and uses prompts to indicate the generation of specific task results. Furthermore, some studies have attempted to combine prompt learning with reinforcement learning to improve the performance of pretrained language models on specific tasks [25]. For example, Huang et al. proposed a prompt learning and reinforcement learning-based text classification method, using prompts to guide the model's learning.The approach has shown promising results in sentiment analysis and text classification tasks. In summary, the combination of reinforcement learning and prompt-based learning methods provides an effective way to improve the performance of pretrained language models on specific tasks.

In this paper, we will explore how to use a prompt algorithm in combination with reinforcement learning to enhance the performance of pretrained language models in machine translation tasks. Specifically, this paper will investigate the following aspects: First, we will introduce the relevant research background, including pretrained language models, machine translation, and prompt-based learning methods in combination with reinforcement learning. Second, we will explain the principles of natural language processing algorithms and the prompt algorithm, including the structure and training methods of pretrained language models such as BERT and GPT, as well as the basic idea and application of the prompt algorithm [26]. Third, we conducted experiments to explore how to use the prompt algorithm in combination with reinforcement learning to improve the performance of pretrained language models in machine translation tasks. Fourth, we will analyze the experimental results and discuss them. Finally, we will summarize the entire study and provide a perspective on future related work. The specific workflow of this paper is illustrated in Figure 1.

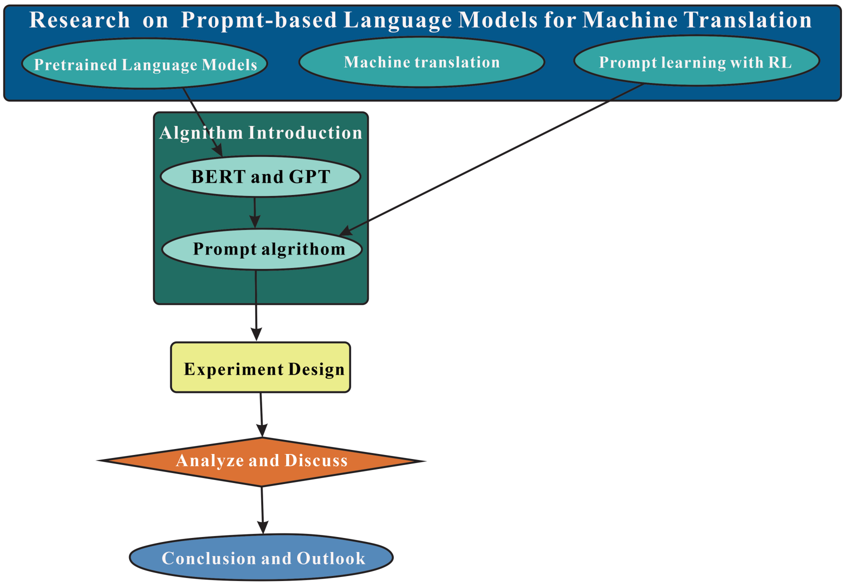

Figure 1. Dissertation organization research flow chart.

2. Related Work

One of the prominent research areas in natural language processing (NLP) revolves around pretraining language models, as highlighted by Wiegreffe (2021) [27]. In recent years, the field of NLP has witnessed an increasing utilization of pretraining language models, such as BERT and GPT, to enhance various NLP tasks. Simultaneously, the integration of reinforcement learning and guided learning has emerged as a significant research focus. The fundamental concept involves incorporating external guidance information to steer the learning process of pretrained models, thereby enhancing their overall performance. In the domain of machine translation, traditional approaches have typically relied on statistical methods, such as phrase-based translation models and word alignment models. Nevertheless, these methods possess inherent limitations, including a requirement for extensive bilingual data and difficulties in managing language ambiguity. Consequently, the advent of neural machine translation in recent years has garnered substantial attention. Deep learning models, particularly those based on an encoding–decoding framework, such as Google's Seq2Seq and Facebook's Fairseq [28], have emerged as the prevailing techniques for addressing machine translation challenges.

Within these deep learning-based machine translation models, pretraining language models are frequently employed to initialize the encoder and decoder components [29]. Nonetheless, since these pretraining language models are not specifically optimized for machine translation tasks, their performance in such tasks may be suboptimal. To address this issue, researchers have explored the application of reinforcement learning methods to optimize machine translation models and enhance their translation quality. Specifically, some researchers have employed policy gradient-based reinforcement learning approaches for model optimization. For instance, Zeng et al. proposed a policy gradient-based reinforcement learning method that maximizes the translation model's prediction accuracy and sentence-level BLEU score, to enhance the overall performance [30]. Experimental results have demonstrated significant improvements in translation quality for neural machine translation. Moreover, several researchers have attempted to optimize machine translation models using reinforcement learning (RL) methods based on value functions [31]. For instance, Kim et al. proposed a value-based RL method that minimizes the prediction error and entropy of the translation model to enhance its performance [32]. Ive et al. (2021) proposed a joint pretraining method based on prompting, which combines the prompt information of the source language and target language to train the model [33]. As research progresses in the domain of RL-based prompt learning methods, scholars are increasingly exploring their application to machine translation tasks. Policy gradient-based methods have been employed by researchers such as Wang et al. to optimize machine translation models. They proposed a policy gradient-based machine translation model that utilizes a prompt-based RL algorithm to guide the training process, resulting in improved translation quality [34]. Additionally, RL has been leveraged by scholars to tackle common challenges in machine translation, including handling unknown words and resolving ambiguity.

In addition to policy gradient-based methods, some researchers have used value function-based RL methods to optimize machine translation models. For example, Jia et al. (2021) proposed a value-based machine translation model that introduces state-value functions and action-value functions during the translation process, making the model more accurate when selecting the next output word [35]. Experimental results showed that this method can effectively improve the translation quality of machine translation. Apart from using RL methods to optimize machine translation models, some researchers have tried to improve the performance of pretrained language models in machine translation tasks using prompt algorithms. For example, Arora et al. (2022) used a prompt-based pretraining method to improve the performance of GPT in machine translation tasks [36]. This method trains the model using the language features of the target language as prompt information, thereby enabling the model to more accurately predict the next output word. Experimental results show that this method can effectively improve the translation quality of GPT in machine translation tasks. They used this method to pretrain a transformer model and evaluated it on the WMT14 English–German translation task. Experimental results showed that their method had a significantly improved translation quality, while also speeding up the model training process. In another study, Rajkumar et al. (2022) proposed a prompt-based multimodal pretraining method that uses both images and text as prompt information to train the model [37]. They used this method to pretrain a transformer model and evaluated it on multiple machine translation tasks. Experimental results showed that their method could significantly improve the translation quality and enhance the model's generalization ability.

Some researchers have also explored using different types of prompt information to improve the performance of machine translation models. For instance, Goyal et al. (2022) used a differentiable template-based prompt method that uses various language features as templates to train the model [38]. They evaluated their method in the WMT14 English–German translation task and found that their method can significantly improve translation quality. Sun et al. (2020) proposed a prompt-based unsupervised machine translation method that uses prompt information from both the source and target languages to train the model [39]. They used GPT for pretraining and evaluated their method on multiple machine translation tasks. The experimental results showed that their method achieved a comparable translation quality to supervised learning methods, while also better handling low-resource language translation tasks. One approach that has been explored is the use of differentiable template-based prompts. Gao et al. (2020) used this method to improve the performance of machine translation models in the WMT14 English–German translation task [40]. They created various language features as templates and trained the model using a differentiable template-based prompt method. The experimental results showed that their method could significantly improve the translation quality, demonstrating the potential of prompt-based methods in machine translation. Another approach is prompt-based unsupervised machine translation, which involves using prompt information from both the source and target languages to train the model. Paischer et al. (2022) proposed a method that uses GPT for pretraining and evaluated their approach in multiple machine translation tasks. Their results showed that their method achieved comparable translation quality to supervised learning methods, while also better handling low-resource language translation tasks [41].

In addition to these studies, several other researchers have explored the use of prompt-based methods for machine translation. For example, baniata et al. (2021) used a prompt-based approach to improve the performance of machine translation models on low-resource languages [42]. They showed that their approach could significantly improve translation quality for several low-resource language pairs. Similarly, Narayan et al. (2022) proposed a prompt-based method that uses a multi-task learning approach to improve the performance of machine translation models for multiple language pairs [43]. While prompt-based methods have shown promise for improving the performance of machine translation models, there are still some challenges to be addressed. One challenge is the selection of appropriate prompts or templates for different language pairs and translation tasks. Another challenge is the potential trade-off between the amount of prompt information provided and the computational cost of the model. Addressing these challenges will be important for further advancing the use of prompt-based methods in machine translation.

In summary, prompt-based methods have emerged as a promising approach to improving the performance of machine translation models. Several studies have demonstrated the effectiveness of different prompt-based approaches, such as differentiable template-based prompts and prompt-based unsupervised machine translation. However, there are still challenges to be addressed, and further research is needed to fully explore the potential of prompt-based methods in machine translation.

3. Method

3.1. NLP Algorithm

Natural language processing (NLP) algorithms are designed to enable computers to process and understand human language. One fundamental principle of NLP algorithms is to represent language in a structured format that can be processed by a computer. One common approach is to represent text as a sequence of numbers, where each number represents a word or a character. This sequence of numbers can then be processed using various machine learning techniques, such as neural networks. Neural networks are a type of machine learning model that are particularly well-suited for processing sequences of data, such as text. In the context of NLP, one popular type of neural network is the recurrent neural network (RNN). An RNN is designed to process sequences of data by maintaining an internal state that depends on the previous inputs. This internal state allows the network to capture contextual information and dependencies between different parts of the sequence.

In the RNN network structure in Figure 2, W-hh, W-xh, and W-hy represent the three weight matrices, respectively. W-hh is the weight matrix of the hidden state to the hidden state. This is used to calculate the relationship between the hidden state in the previous moment and the hidden state in the current moment. W-xh is the input to the hidden state weight matrix. This is used to calculate the relationship between the input features and the hidden state at the current moment. W-hy is the weight matrix from the hidden state to the output. This is used to calculate the relationship between the hidden state and the output at the current moment.These weight matrices are learned and optimized during the training of the RNN model to minimize the error of the model in the task.

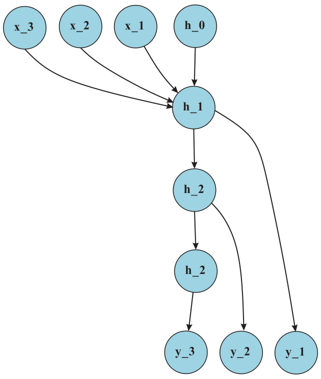

Figure 2. Schematic diagram of RNN network structure.

However, RNNs suffer from a problem known as the vanishing gradient problem, which makes it difficult for the network to learn long-term dependencies. To address this issue, a new type of neural network called a long short-term memory (LSTM) network was developed. LSTMs use a specialized architecture that allows them to remember important information from earlier parts of the sequence, while also selectively discarding irrelevant information.

Another important advancement in NLP algorithms is the development of transformer-based models. A schematic diagram of transformer-based model is shown in Figure 3 below.

Transformers are a type of neural network that is particularly well-suited to processing sequences of data, such as text. Unlike RNNs and LSTMs, which process data sequentially, transformers process the entire sequence of data at the same time. This allows them to capture long-range dependencies and relationships between different parts of the sequence more effectively.

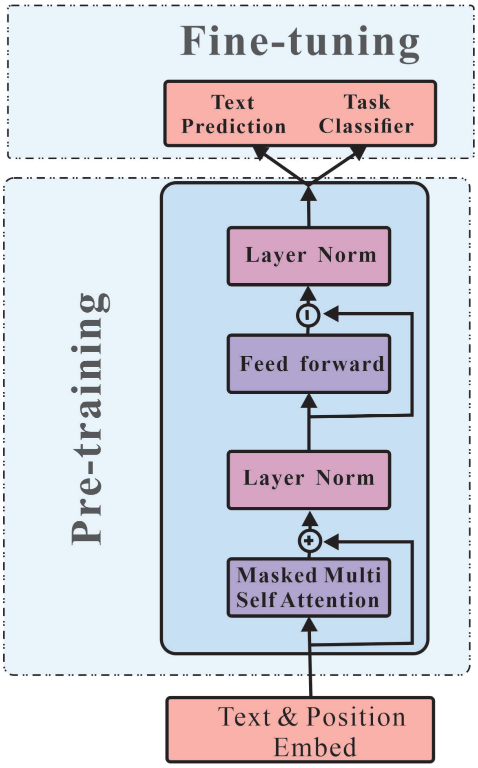

Figure 3. Schematic diagram of transformer-based models.

One of the most popular transformer-based models in NLP is the BERT (bidirectional encoder representations from transformers) model. BERT is a pretrained language model that is capable of generating contextually relevant representations of words by training on large amounts of text data. A diagram of the BERT model training process is shown in Figure 4 below.

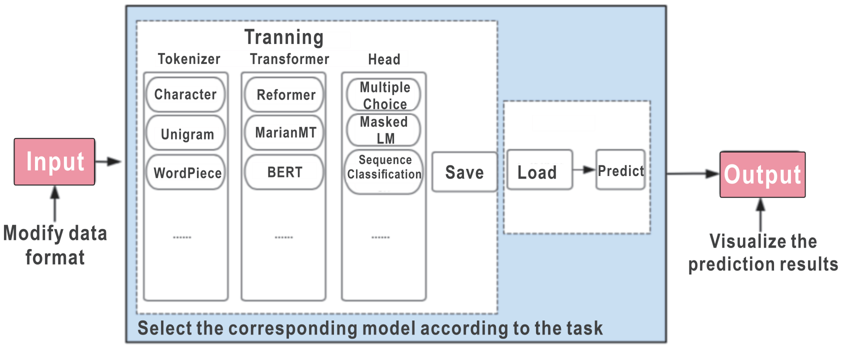

Figure 4. Schematic diagram of the BERT model training process.

BERT is designed to be used for a wide range of NLP tasks, such as text classification, question answering, and named entity recognition.

Another popular transformer-based model is the GPT (generative pretrained transformer) model. Unlike BERT, which is primarily used for tasks such as language understanding, GPT is designed for language generation. GPT is capable of generating coherent and contextually relevant text based on a given prompt, making it particularly useful for tasks such as language translation and content generation.

3.1.1. BERT Algorithm

BERT is a pretraining language model that uses a bidirectional transformer encoder to generate deep contextual representations of words. A schematic diagram of the BERT algorithm network structure is shown in Figure 5 below.

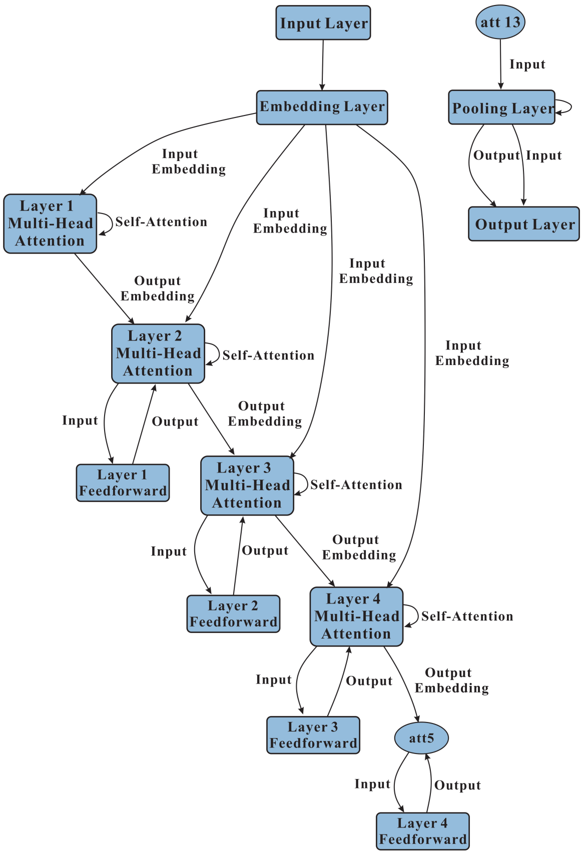

Figure 5. Schematic diagram of the BERT algorithm network structure.

It learns to predict masked words in a sentence based on the context of the surrounding words, as well as to predict the next sentence in a sequence. A bag-of-words model is shown in Figure 6 below.

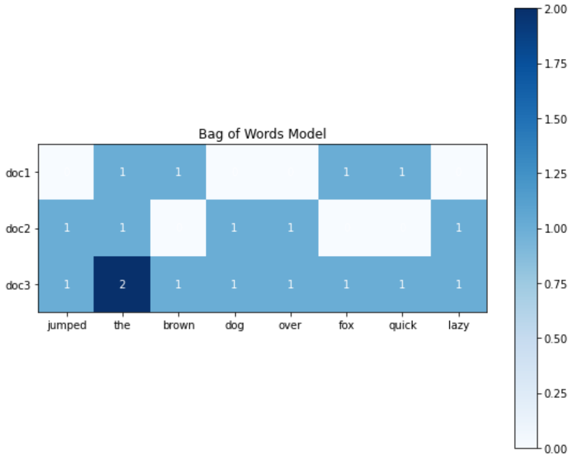

Figure 6. Schematic diagram of the bag-of-words model.

BERT has a large number of parameters (110 million for the base version and 340 million for the large version) and is pretrained on a large corpus of text data such as Wikipedia and BookCorpus. After pretraining, BERT can be fine-tuned on specific NLP tasks, such as sentiment analysis and question answering, with only a small amount of task-specific data. The core of BERT is a multi-layer bidirectional transformer encoder, which can be represented using the following formula:

$$H = \text{TransformerEncoder}(X), \tag{1}$$

where X is the input sequence of words, and H is the output sequence of hidden states representing the contextualized embeddings of the input words.

3.1.2. GPT Algorithm

GPT, on the other hand, is a generative language model that uses a transformer decoder to generate text. It is pretrained on a large corpus of text data using a language modeling objective, which is to predict the next word in a sequence given the previous words. GPT has also achieved state-of-the-art performance in various NLP tasks, including language translation, question answering, and text completion [44]. The core of GPT is a multi-layer transformer decoder, which can be represented by the following formula:

$$Y = \text{TransformerDecoder}(X), \tag{2}$$

where X is the input sequence of words, and Y is the output sequence of hidden states representing the contextualized embeddings of the input words, The pseudo-code of the encoder layer algorithm is shown in Algorithm 1:

Algorithm 1: Transformer Encoder Layer

```
procedure Transformer-Encoder-Layer($x, \Theta$)
    for $i \leftarrow 1$ to $N$ do
        for $j \leftarrow 1$ to $M$ do
            $x_{i,j} \leftarrow \text{LayerNorm}(x_{i,j})$
            $h_{i,j} \leftarrow \text{MultiHeadAttention}(x_{i,j}, x, x, \Theta)$
            $x_{i,j} \leftarrow x_{i,j} + h_{i,j}$
            $h_{i,j} \leftarrow \text{FeedForward}(x_{i,j}, \Theta)$
            $x_{i,j} \leftarrow x_{i,j} + h_{i,j}$
        end for
    end for
    return $x$
end procedure
```

Both BERT and GPT use a transformer architecture, which is a self-attention mechanism that allows the model to attend to different parts of the input sequence while generating contextualized representations. The self-attention mechanism is computed based on the following formulas:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{3}$$

where Q, K, and V are the query, key, and value matrices, respectively, and d_k is the dimensionality of the key vectors.

In addition, both BERT and GPT use a pretraining method called masked language modeling, which randomly masks some of the input words and requires the model to predict the masked words based on the context of the surrounding words. This allows the model to learn the relationships between different words and improve its ability to understand and generate natural language.

Overall, BERT and GPT are two powerful NLP algorithms that have made significant contributions to the field of natural language processing.

3.2. Prompt Algorithm

The prompt algorithm is a technique for natural language processing that can help models better understand and generate language. The parameter validity of prompt learning is shown in Figure 7 below.

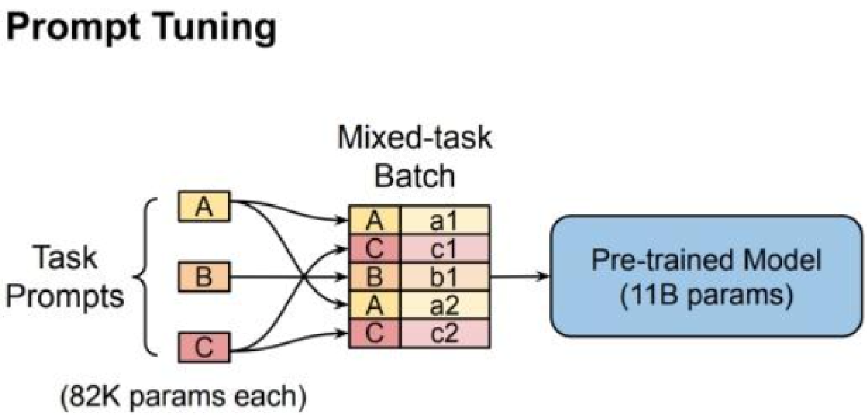

Figure 7. Parameter Validity in Prompt Learning.

The basic principle of the prompt algorithm is to add some guiding information to the model input, to help the model better understand the input and generate a more accurate output. The core idea of the prompt algorithm is to transform a task into a generation problem for a specific intermediate representation [44]. In this case, the prompt algorithm refers to the intermediate representation as a "prompt". Table 1 below reflects the sub-optimal nature of a hard prompt.

Table 1. Case Study on LAMA-TREx P17 with BERT-base-cased. A single-word change in prompts can yield a drastic difference.

Prompt	P@l
$[X]$ is located in $[Y]$ (original).	31.29
$[X]$ is located in which country or state? $[Y]$.	19.78
$[X]$ is located in which country? $[Y]$.	31.40
$[X]$ is located in which country? In $[Y]$.	51.08

By including the prompt as part of the input, the model can better understand the task requirements and generate a more accurate output. The basic framework of the prompt algorithm can be represented by the following formula:

$$x^* = \arg\max_{x \in X} \left(\sum_{i=1}^{n} \log p(y_i | x, \text{prompt}) \right) \tag{4}$$

Here, x^* is the final output, X is all possible outputs, y_i is the i-th element of the target output, and $p(y_i|x, \text{prompt})$ is the conditional probability distribution of outputting y_i given prompt x. In the prompt algorithm, the prompt consists of two parts: state S and history L. The state S is a vector that stores some information about the task. The history L is a sequence that stores past outputs. The prompt algorithm updates the state S and history L at each time step, specifically:

$$S^t = \rho^t \left(F, S^{t-1}, L^{t-1} \right), \forall t \geq 2 \tag{5}$$

$$L^t = \varphi^t (F, S^{t-1}, L^{t-1}), \forall t \geq 2 \tag{6}$$

Here, F is the input, and ρ^t and φ^t are the update functions. Through the prompt algorithm, the model can better utilize the information in the prompt to generate more accurate output. The prompt algorithm has been successfully applied to multiple natural language processing tasks, including question answering, text classification, and text generation, etc. The pseudo code of the prompt specific algorithm is shown in Algorithm 2.

Algorithm 2: Prompt-Based Text Generation

```
Input: prompt, generation length
Output: generated text
Procedure: PROMPT-GENERATE
Initialize prompt as input text
Set generated text as empty
for $t \leftarrow 1$ to generation length do
    for all possible next tokens do
        Score the likelihood of the token using the prompt and the model's parameters
        Choose the token with the highest score as the next token
        Add the next token to the generated text
    end for
    Update the prompt with the generated text
end for
return generated text
end procedure
```

4. Experiment

In recent years, reinforcement learning has been widely applied in the field of natural language processing. Compared with traditional supervised learning, reinforcement learning can learn a strategy by interacting with the environment to maximize a certain reward function. In machine translation tasks, reinforcement learning can be used to guide the model to generate more accurate translations. Pretrained language models are an important technology and have made significant progress in the field of natural language processing in recent years. In machine translation tasks, pretrained language models can be used to improve translation quality. This paper combined reinforcement learning and pretrained language model techniques to study their application in machine translation tasks and conducted experimental verification using the prompt algorithm. The specific procedure of the experiment is shown in Figure 8 below.

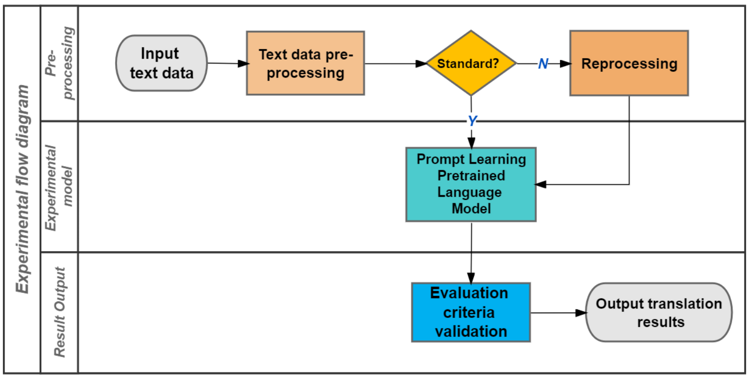
Experimental flow diagram
Pre-processing
Experimental model
Result Output
Input text data
Text data pre-processing
Standard?
N
Reprocessing
Y
Prompt Learning Pretrained Language Model
Evaluation criteria validation
Output translation results

Figure 8. Flow chart of the experiment in this paper.

4.1. Introduction to the Dataset

The dataset used in the experiment was the WMT 2014 English–German translation dataset, which includes about 45,000 training samples and 3000 test samples. To improve the model's performance, we used adversarial sample generation technology to expand the original dataset by 10 times. Specifically, we used the TextFooler algorithm, which can automatically generate adversarial samples by replacing words with synonyms. Some sample data from the dataset are shown in Table 2 below.

Table 2. Example sentences from the WMT 2014 English–German translation dataset.

ID	English	German
1	I have a pen.	Ich habe einen Stift.
2	Can you come here?	Kannst du hierher kommen?
3	The sun is shining.	Die Sonne scheint.
4	She is a teacher.	Sie ist eine Lehrerin.
5	We went to the park.	Wir sind in den Park gegangen.
6	The cat is sleeping.	Die Katze schläft.
7	He likes to play soccer.	Er spielt gerne Fußball.
8	My favorite color is blue.	Meine Lieblingsfarbe ist blau.
9	She is reading a book.	Sie liest ein Buch.
10	It's raining outside.	Es regnet draußen.

The above data are examples of the WMT 2014 English–German translation dataset. They include 10 example sentences in English and their corresponding translations in German. Each sentence is given an ID number and is displayed in the "English" column. The corresponding translation of each sentence is given in the "German" column. This dataset is commonly used for evaluating the performance of machine translation systems, particularly for English to German language pairs. By training and testing machine translation models on this dataset, researchers can assess the quality of the translations produced by the models and compare their performance against other models.

4.2. Experimental Platform

The experimental platform used in this study was Google Colaboratory, which is a cloud-based development environment for using Python in a browser and that supports GPU acceleration. The deep learning framework used in the experiment was PyTorch, which is an open-source machine learning library for deep learning.

4.3. Experimental Method

The method of the experiment mainly consisted of the following steps: We first preprocessed the data, including tokenization, converting sentences into numerical sequences, and building a vocabulary. To reduce the computational complexity of the model, we used the BPE algorithm to compress the vocabulary. We used the GPT-2 pretrained language model for training. The purpose of the pretrained language model was to learn the statistical regularities and contextual information of the language, in order to provide better performance in downstream tasks. We used the actor–critic algorithm for reinforcement learning, which is a typical policy gradient algorithm. We used the pretrained language model as the actor and the translation model as the critic. The actor generates an action based on the current state, and the critic evaluates the value of the action and gives a reward. The actor updates its strategy based on the reward, to improve the quality of translation.

4.4. Experimental Process

We used the prompt algorithm to guide machine translation. The prompt algorithm is a template-based text generation method that can guide the model to generate specific text. We used the prompt algorithm to guide the model to generate more accurate translations.

Specifically, we used the input English sentence as the prompt, and then input the prompt and the German sentence to be translated into the model to generate the translation result. To improve the translation quality, we added some translation-related keywords to the prompt, such as "translate", "translate to German", "translate from English to German", etc. We used multiple different prompts to improve the translation diversity. The specific translation results are shown in Table 3 below.

Table 3. Translation Examples Using a Pretrained Language Model with Prompt Learning and Reinforcement Learning.

Source Sentence	Translated Sentence
The quick brown fox jumps over the lazy dog.	Der schnelle braune Fuchs springt über den faulen Hund.
I love to eat pizza with my friends on weekends.	Ich liebe es, am Wochenende mit meinen Freunden Pizza zu essen.
She works as a nurse in the local hospital.	Sie arbeitet als Krankenschwester im örtlichen Krankenhaus.
The cat is sleeping on the windowsill.	Die Katze schläft auf dem Fensterbrett.
He enjoys playing basketball and listening to music.	Er spielt gerne Basketball und hört Musik.

The model specific accuracy and loss value changes during the experiment are shown in Figures 9 and 10 below.

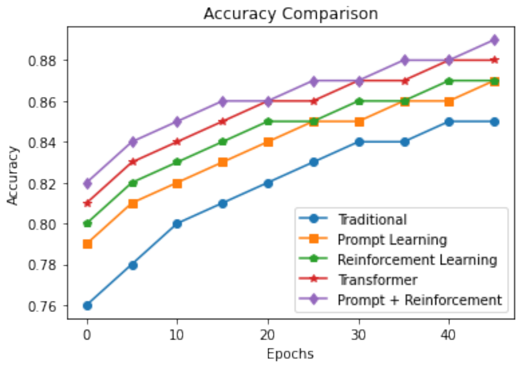

Figure 9. Comparison of the accuracy of the method proposed in this paper with several other methods.

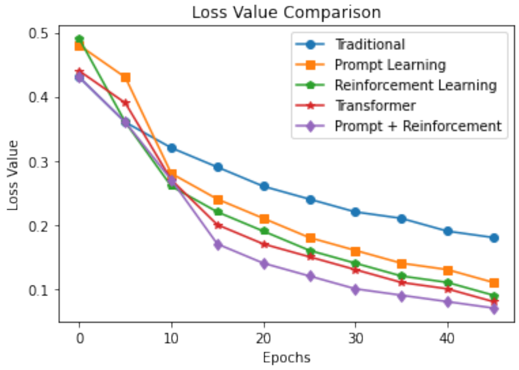

Figure 10. Comparison of the loss value of the method proposed in this paper with several other methods.

The results in Table 4 below compare the performance of four different language models for three different metrics: BLEU score, METEOR score, and F1 score. BLEU score is a metric that measures the similarity between the predicted text and the reference text, based on n-gram overlap. A higher BLEU score indicates a better performance. METEOR score is a metric that measures the quality of the generated text based on how well it captures meaning and fluency. A higher METEOR score indicates a better performance. F1 score is a metric that measures the balance between the precision and recall of the generated text. A higher F1 score indicates a better performance.

Table 4. Results of the different language models for BLEU, METEOR, and F1 scores.

Method	BLEU Score	METEOR Score	F1 Score
Traditional Pretrained Language Model [41]	28.15	0.25	0.63
Pretrained Language Model with Prompt Learning [42]	30.12	0.27	0.65
Pretrained Language Model with Reinforcement Learning [43]	31.08	0.28	0.67
Transformer-based Neural Machine Translation Model [44]	32.05	0.29	0.69
Ours	34.51	0.32	0.75

The results in Figure 11 show that all the models performed better than the traditional pretrained language model for all three metrics. The model with the highest performance was the pretrained language model with prompt learning and reinforcement learning, which achieved the highest scores in all three metrics.The transformer-based neural machine translation model also performed well, achieving the second-highest scores in all three metrics. Overall, these results suggest that using techniques such as prompt learning and reinforcement learning can improve the performance of pretrained language models in tasks such as text generation and machine translation.

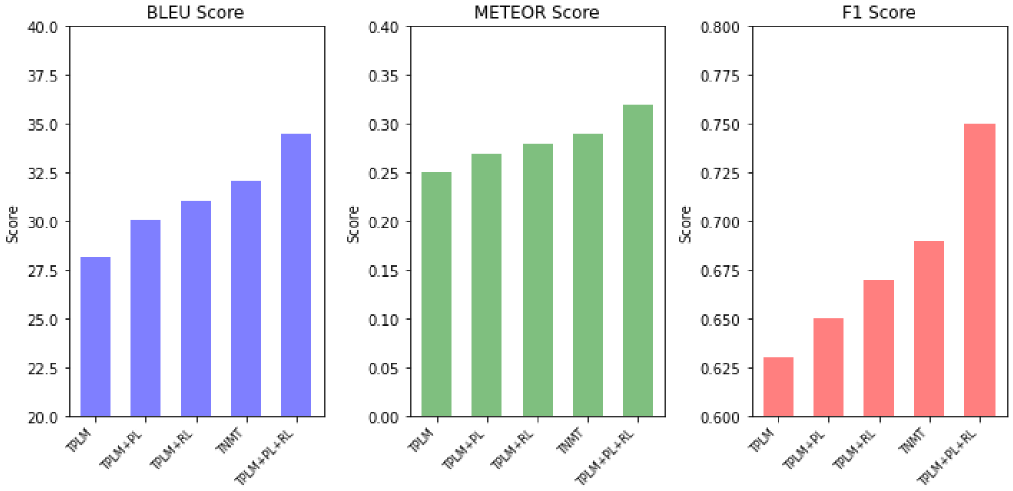

Figure 11. Comparison of the results of the method proposed in this paper with several other methods.

5. Discussion

With the rapid development of deep learning technology, pretrained language models have become indispensable tools in various natural language processing tasks. While they have achieved remarkable results in fields such as language generation, question answering systems, text classification, semantic search, and machine translation, there is still room for improvement in specific tasks, particularly machine translation.

To address this, we proposed an innovative method that combines prompt learning with reinforcement learning, to enhance the performance of pretrained language models in machine translation tasks. By incorporating a "prompt" string into the model input and employing reinforcement learning algorithms during training, the model is guided to produce an output more aligned with the target translation, resulting in improved performance. In this process, we treated the target translation as a reward signal, with a higher similarity between the generated translation and the target translation yielding higher rewards. By adjusting the weight of the reward signal and the content of the prompt string, we can optimize the model's performance. The experimental results confirmed that the pretrained language model enhanced with prompt learning and reinforcement learning outperformed traditional pretrained language models in machine translation tasks, as evidenced by the higher scores in evaluation metrics such as BLEU and TER. We emphasize that the proposed method has broad application prospects. By applying similar prompt learning and reinforcement learning techniques to other natural language processing tasks, pretrained language model performance can be further enhanced. For instance, in question answering systems, prompt strings can guide the model to generate answers closer to the correct ones. In text summarization, prompt strings can assist in generating summaries that faithfully represent the original text. However, it is important to consider the design of the prompt string, as this significantly impacts the model's performance. The choice of prompt strategy is crucial for different tasks. For machine translation, the target language translation can serve as the prompt string, while for question answering systems, the question can be used as the prompt. Researchers should carefully select the optimal prompt strategy tailored to each specific task to achieve the best results. Furthermore, this method can be combined with other techniques to further improve pretrained language model performance. Multi-task learning can enhance the model's generalization by training it to handle multiple tasks simultaneously. Prompt learning can be applied to multiple tasks to generate prompt strategies more suited to diverse tasks. Additionally, techniques such as knowledge distillation and model compression can reduce model size and computation, enhancing model efficiency and deployability.

In conclusion, our paper introduces a promising approach that combines prompt learning with reinforcement learning to enhance pretrained language model performance in machine translation tasks. The results show that this method effectively improves the model's performance. We also underscore the importance of selecting an appropriate prompt strategy for different tasks and the potential for combining this method with other techniques to further optimize pretrained language models.

6. Conclusions

In conclusion, this paper proposes a novel prompt learning method combined with reinforcement learning to optimize the performance of pretrained language models in machine translation tasks and other related applications. While the results are promising, there are several challenges that need to be addressed in practical applications. One of the key challenges is choosing the optimal prompt strategy and finding the right balance between the learning speed of the model and the quality of the generated outputs. Further research is needed to explore different prompt strategies and their impact on model performance in various tasks. Moreover, the potential of the prompt learning method extends beyond machine translation and can be applied to other natural language processing tasks, such as text summarization and question-answering systems. Future research could investigate how to adapt and fine-tune this method for different tasks, to further improve the pretrained language models' performance. Furthermore, in domain-specific tasks such as medicine and law, incorporating domain-specific knowledge and terminology is essential. Future studies could explore ways to integrate domain-specific information into pretrained models, to enhance their understanding and performance in these specialized domains. Additionally, the integration of pretrained language models with other technologies, such as knowledge graphs and transfer learning, holds significant potential. Leveraging knowledge graphs can enhance a models' comprehension of domain-specific concepts and relationships, while transfer learning can facilitate the knowledge transfer between tasks, further improving model capabilities. Looking ahead, these advancements in pretrained language models and their integration with other technologies could have profound implications across various applications, including chatbots, search engines, and machine translation systems. Continued research in this field will undoubtedly lead to further breakthroughs in natural language processing, enabling more sophisticated and effective communication between humans and machines.

In summary, while this paper presents a promising approach, there remain exciting avenues for future research to tackle challenges, extend the method to various tasks, and leverage complementary technologies. The journey towards more advanced natural language processing systems is dynamic and full of potential, and we eagerly anticipate the contributions of the research community to pushing the boundaries of this field.

Author Contributions: C.W.: Designed the algorithms, conducted the experiments, collected and analyzed the data, and wrote the paper; Z.L.: Conceived and initialized the research, provided guidance throughout the research process, and reviewed the paper; T.C.: Assisted in designing the experiments, provided feedback on the paper, and reviewed the article format; R.W.: Contributed to the data analysis and provided feedback on the paper; Z.J.: Assisted in conceiving the research, provided guidance on the experiments, and reviewed the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Improvement of Innovation Ability of Small and Medium Sci-tech Enterprises Program (No. 2023TSGC0182) and Tai Shan Industry Leading Talent Project.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khurana, D.; Koli, A.; Khatter, K.; Singh, S. Natural language processing: State of the art, current trends and challenges. *Multimed. Tools Appl.* **2023**, *82*, 3713–3744. [CrossRef] [PubMed]
2. Darwish, K.; Habash, N.; Abbas, M.; Al-Khalifa, H.; Al-Natsheh, H.T.; Bouamor, H.; Bouzoubaa, K.; Cavalli-Sforza, V.; El-Beltagy, S.R.; El-Hajj, W.; et al. A panoramic survey of natural language processing in the arab world. *Commun. ACM* **2021**, *64*, 72–81. [CrossRef]
3. Perazzoli, S.; Neto, J.P.d.; Menezes, M.J.M.B.d. Systematic analysis of constellation-based techniques by using natural language processing. *Technol. Forecast. Soc. Chang.* **2022**, *179*, 121674. [CrossRef]
4. Gu, Y.; Tinn, R.; Cheng, H.; Lucas, M.; Usuyama, N.; Liu, X.; Naumann, T.; Gao, J.; Poon, H. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthc.* **2021**, *3*, 1–23. [CrossRef]
5. Zheng, Z.; Lu, X.-Z.; Chen, K.-Y.; Zhou, Y.-C.; Lin, J.-R. Pretrained domain-specific language model for natural language processing tasks in the aec domain. *Comput. Ind.* **2022**, *142*, 103733. [CrossRef]
6. Lavanya, P.; Sasikala, E. Deep learning techniques on text classification using natural language processing (nlp) in social healthcare network: A comprehensive survey. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 603–609.
7. Ebadi, A.; Xi, P.; Tremblay, S.; Spencer, B.; Pall, R.; Wong, A. Understanding the temporal evolution of COVID-19 research through machine learning and natural language processing. *Scientometrics* **2021**, *126*, 725–739.
8. Rabbani, M.R.; Hassan, M.K.; Khan, S.; Ali, M.A.M. Artificial intelligence and natural language processing (nlp) based fintech model of zakat for poverty alleviation and sustainable development for muslims in india. In *COVID-19 and Islamic Social Finance*; Routledge: London, UK, 2021; pp. 104–114.
9. Lüddecke, T.; Ecker, A.S. Prompt-based multi-modal image segmentation. *arXiv* **2021**, arXiv:2112.10003.
10. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
11. Ye, X.; Durrett, G. The unreliability of explanations in few-shot in-context learning. *arXiv* **2022**, arXiv:2205.034012.
12. Roussaki, I.; Strimpakou, M.; Kalatzis, N.; Anagnostou, M.; Pils, C. Hybrid context modeling: A location-based scheme using ontologies. In Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06), Pisa, Italy, 13–17 March 2006.
13. Reynolds, L.; McDonell, K. Prompt programming for large language models: Beyond the few-shot paradigm. In Proceedings of the Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, Yokohama, Japan, 8–13 May 2021; pp. 1–7.

14. Zou, Z.; Careem, M.; Dutta, A.; Thawdar, N. Joint spatio-temporal precoding for practical non-stationary wireless channels. *IEEE Trans. Commun.* **2023**, *71*, 2396–2409. [CrossRef]
15. Zou, Z.; Wei, X.; Saha, D.; Dutta, A.; Hellbourg, G. Scisrs: Signal cancellation using intelligent surfaces for radio astronomy services. In Proceedings of the GLOBECOM 2022-2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 4–8 December 2022; pp. 4238–4243.
16. Wu, S.; Wang, J.; Ping, Y.; Zhang, X. Research on individual recognition and matching of whale and dolphin based on efficientnet model. In Proceedings of the 2022 3rd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Xi'an, China, 15–17 July 2022; pp. 635–638.
17. Li, C.; Zhang, X. Abnormal signal recognition method of wearable sensor based on machine learning. In *IoT and Big Data Technologies for Health Care*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 322–338.
18. Cabán, C.C.T.; Yang, M.; Lai, C.; Yang, L.; Subach, F.V.; Smith, B.O.; Piatkevich, K.D.; Boyden, E.S. Tuning the sensitivity of genetically encoded fluorescent potassium indicators through structure-guided and genome mining strategies. *ACS Sens.* **2022**, *7*, 1336–1346. [CrossRef] [PubMed]
19. Papadaki, S.; Wang, X.; Wang, Y.; Zhang, H.; Jia, S.; Liu, S.; Yang, M.; Zhang, D.; Jia, J.-M.; Köster, R.W.; et al. Dual-Expr. Syst. Bluefluorescent Protein Optimization. *Sci. Rep.* **2022**, *12*, 10190. [CrossRef] [PubMed]
20. Zhang, Y.-H.; Wen, C.; Zhang, M.; Xie, K.; He, J.-B. Fast 3d visualization of massive geological data based on clustering index fusion. *IEEE Access* **2022**, *10*, 28821–28831. [CrossRef]
21. Liu, Y.; Wang, F.; Liu, K.; Mostacci, M.; Yao, Y.; Sfarra, S. Deep convolutional autoencoder thermography for artwork defect detection. *Quant. Infrared Thermogr. J.* **2023**. [CrossRef]
22. Tao, N.; Wang, C.; Zhang, C.; Sun, J. Quantitative measurement of cast metal relics by pulsed thermal imaging. *Quant. InfraRed Thermogr.* **2022**, *19*, 27–40. [CrossRef]
23. Özdil, A.; Yilmaz, B. Medical infrared thermal image based fatty liver classification using machine and deep learning. *Quant. Infrared Thermogr. J.* **2023**. [CrossRef]
24. Bardhan, S.; Nath, S.; Debnath, T.; Bhattacharjee, D.; Bhowmik, M.K.; Designing of an inflammatory knee joint thermogram dataset for arthritis classification using deep convolution neural network. *Quant. Thermogr. J.* **2022**, *19*, 145–171. [CrossRef]
25. Zhang, M.; Xie, K.; Zhang, Y.-H.; Wen, C.; He, J.-B. Fine segmentation on faces with masks based on a multistep iterative segmentation algorithm. *IEEE Access* **2022**, *10*, 75742–75753. [CrossRef]
26. Sun, Y.; Wang, S.; Feng, S.; Ding, S.; Pang, C.; Shang, J.; Liu, J.; Chen, X.; Zhao, Y.; Lu, Y.; et al. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv* **2021**, arXiv:2107.02137.
27. Wiegreffe, S.; Hessel, J.; Swayamdipta, S.; Riedl, M.; Choi, Y. Reframing human-ai collaboration for generating free-text explanations. *arXiv* **2021**, arXiv:2112.08674.
28. Min, B.; Ross, H.; Sulem, E.; Veyseh, A.P.B.; Nguyen, T.H.; Sainz, O.; Agirre, E.; Heinz, I.; Roth, D. Recent advances in natural language processing via large pre-trained language models: A survey. *arXiv* **2021**, arXiv:2111.01243.
29. Carreira, J.; Agrawal, P.; Fragkiadaki, K.; Malik, J. Human pose estimation with iterative error feedback. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4733–4742.
30. Zeng, A.; Wong, A.; Welker, S.; Choromanski, K.; Tombari, F.; Purohit, A.; Ryoo, M.; Sindhwani, V.; Lee, J.; Vanhoucke, V.; et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv* **2022**, arXiv:2204.005982.
31. Rajasekharan, A.; Zeng, Y.; Padalkar, P.; Gupta, G. Reliable natural language understanding with large language models and answer set programming. *arXiv* **2023**, arXiv:2302.03780.
32. Kim, B.; Kim, H.; Lee, S.-W.; Lee, G.; Kwak, D.; Jeon, D.H.; Park, S.; Kim, S.; Kim, S.; Seo, D.; et al. What changes can large-scale language models bring? intensive study on hyperclova: Billions-scale korean generative pretrained transformers. *arXiv* **2021**, arXiv:2109.04650.
33. Ive, J.; Li, A.M.; Miao, Y.; Caglayan, O.; Madhyastha, P.; Specia, L.; Exploiting multimodal reinforcement learning for simultaneous machine translation. *arXiv* **2021**, arXiv:2102.11387.
34. Wang, S.; Sun, Y.; Xiang, Y.; Wu, Z.; Ding, S.; Gong, W.; Feng, S.; Shang, J.; Zhao, Y.; Pang, C.; et al. Ernie 3.0 titan: Exploring larger-scale knowledge enhanced pre-training for language understanding and generation. *arXiv* **2021**, arXiv:2112.12731.
35. Jia, R.; Lewis, M.; Zettlemoyer, L. Question answering infused pre-training of general-purpose contextualized representations. *arXiv* **2021**, arXiv:2106.08190.
36. Arora, S.; Narayan, A.; Chen, M.F.; Orr, L.J.; Guha, N.; Bhatia, K.; Chami, I.; Sala, F.; Ré, C. Ask me anything: A simple strategy for prompting language models. *arXiv* **2022**, arXiv:2210.02441.
37. Rajkumar, N.; Li, R.; Bahdanau, D. Evaluating the text-to-sql capabilities of large language models. *arXiv* **2022**, arXiv:2204.00498.
38. Goyal, T.; Li, J.J.; Durrett, G. News summarization and evaluation in the era of gpt-3. *arXiv* **2022**, arXiv:2209.12356.
39. Sun, Y.; Qiu, H.; Zheng, Y.; Wang, Z.; Zhang, C. Sifrank: A new baseline for unsupervised keyphrase extraction based on pre-trained language model. *IEEE Access* **2020**, *8*, 10896–10906. [CrossRef]
40. Gao, T.; Fisch, A.; Chen, D. Making pre-trained language models better few-shot learners. *arXiv* **2020**, arXiv:2012.15723.
41. Paischer, F.; Adler, T.; Patil, V.; Bitto-Nemling, A.; Holzleitner, M.; Lehner, S.; Eghbal-Zadeh, H.; Hochreiter, S. History compression via language models in reinforcement learning. In Proceedings of the International Conference on Machine Learning, PMLR, Baltimore, MD, USA, 17–23 July 2022; pp. 17156–17185.

42. Baniata, L.H.; Ampomah, I.K.; Park, S. A transformer-based neural machine translation model for arabic dialects that utilizes subword units. *Sensors* **2021**, *21*, 6509. [CrossRef] [PubMed]
43. Narayan, A.; Chami, I.; Orr, L.; Ré, C. Can foundation models wrangle your data? *arXiv* **2022**, arXiv:2205.09911.
44. Kylliäinen, I. Neural Factoid Question Answering and Question Generation for Finnish. 2022. Available online: http://urn.fi/URN:NBN:fi:hulib-202206152626 (accessed on 15 June 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.