Research Article

A Study on the Intelligent Translation Model for English Incorporating Neural Network Migration Learning

Yanbo Zhang

School of Foreign Languages, Zhengzhou Shengda University, Xinzheng of Zhengzhou City, 451191, Henan Province, China

Correspondence should be addressed to Yanbo Zhang; yanbozhang@shengda.edu.cn

Received 11 October 2021; Revised 25 October 2021; Accepted 27 October 2021; Published 18 November 2021

Academic Editor: Xin Ning

Copyright © 2021 Yanbo Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Under the current artificial intelligence boom, machine translation is a research direction of natural language processing, which has important scientific research value and practical value. In practical applications, the variability of language, the limited capability of representing semantic information, and the scarcity of parallel corpus resources all constrain machine translation towards practicality and popularization. In this paper, we conduct deep mining of source language text data to express complex, high-level, and abstract semantic information using an appropriate text data representation model; then, for machine translation tasks with a large amount of parallel corpus, I use the capability of annotated datasets to build a more effective migration learning-based end-to-end neural network machine translation model on a supervised algorithm; then, for machine translation tasks with parallel corpus data resource-poor language machine translation tasks, migration learning techniques are used to prevent the overfitting problem of neural networks during training and to improve the generalization ability of end-to-end neural network machine translation models under low-resource conditions. Finally, for language translation tasks where the parallel corpus is extremely scarce but monolingual corpus is sufficient, the research focuses on unsupervised machine translation techniques, which will be a future research trend.

1. Introduction

With the rapid development of the Internet as well as information technology, the field of artificial intelligence is gaining more and more attention, attracting a large number of researchers and developers. Machine translation is a hot spot of research in the field of artificial intelligence, with important theoretical significance and great application value. Machine translation is a research direction of natural language processing, which has important scientific research value and practical value. In the process of translation and processing of massive information, machine translation technology (machine translation) plays a crucial role [1]. Specifically, machine translation can help users deal with a large amount of text, and after users have obtained a large amount of text, they can input the obtained text into the machine translation system and the machine translation system will generate the translation needed by users after judgment and calculation according to the text inputted by users. However, the machine translation is still inadequate compared with

humans; if the user has different views on the machine translation or is not satisfied with the translation, the user can modify the machine translation by himself and finally get a satisfactory translation. For example, if we encounter a text in English in real life, we can input the obtained text into the machine translation system and translate the text with the help of the machine and then check whether the translation is correct and whether it can play a certain role in certain specific occasions after the machine translation is finished.

The traditional approach to solve cross-lingual text matching can be done by first building a translation model and then training the corresponding text matching model after the translation is completed, but this process is often labor intensive, and the crosslingual text translation problem itself is not overly concerned with translation quality. It is also possible to construct matching models directly using traditional methods based on grammars, keywords, sentence patterns, and other statistical linguistic data or semantic letters, but this approach also has some shortcomings. The effect of text matching based on statistical linguistic models

relies heavily on manually designed feature engineering and is computationally intensive, and manually constructed feature engineering is also time-consuming [2]. It is difficult to transfer the model knowledge to other languages to assist learning. However, English, especially the same language family, often has related laws and commonalities, because human languages themselves are constantly interacting with each other and accurately describing and using the common laws of different languages can be very helpful for crosslanguage model learning. Due to the complexity of natural languages and the different degrees of research progress in different languages, it is usually necessary to retrain the models when applying text matching models to different corpora or languages, which is labor intensive, and the matching models trained for some corpora often have limited generalization ability and are not well suited for practical application scenarios [3]. With insufficient training data in the low-resource case, it is difficult for neural networks to train models with easy convergence, high stability, and good generalization ability. The domain-adaptive approach of migration learning, on the other hand, can use the high-resource parallel corpus data to extract the useful information that may be used in the low-resource parallel corpus learning. The domain-adaptive approach of migration learning can use the high-resource parallel corpus data to extract the useful information that may be used in low-resource parallel corpus learning. Therefore, this paper proposes an intelligent English translation model based on neural network migration learning, which can construct feature mapping relations among pretrained language models and search for matching relations in the high-dimensional feature space, which makes good use of the common relations among languages and can use the existing knowledge space to save the resources required for model learning, so it has important research significance and practical value.

2. Related Works

Mainstream neural network models that can be used by end-to-end neural network machine translation systems based on sequence-to-sequence transformation models, encoders, and decoders include recurrent neural networks and improved long- and short-term memory networks and gated recurrent networks, etc.; convolutional neural network models; and translation models based on attention mechanisms. Recently, a large number of other researchers have used reinforcement learning ideas and migratory learning models such as generative adversarial networks for machine translation tasks.

A multilayer convolutional neural network called ConvNet was first proposed in the literature [4], which successfully solved the recognition task related to a grayscale handwritten digital language using a supervised training approach. Based on this, an improved version of ConvNet, the well-known LeNet-5, was proposed in the literature [5], which was applied to optical character recognition tasks with milestone success. In [6], an AlexNet model with five convolutional layers and three fully connected layers was designed, which substantially outperformed the traditional approach

on a million-dollar ImageNet dataset, from more than 70% to more than 80%, and the model won the ILSVRC crown with a top 5 error rate of 16.4%, a significant advantage over the second-place result of 26.2%. AlexNet opened a milestone for deep learning in computer vision, and since then, ILSVRC has been continuously topped by deep learning every year. The literature [7] proposed the VGG model, still consisting of convolutional and fully connected layers, which has a very consistent network structure, using all 3×3 convolutions and 2×2 pooling from start to finish; however, it suffers from consuming more computational resources and using more parameters, resulting in more memory usage. Later, the literature [8, 9] investigated deeper network structures based on AlexNet, which is no longer limited to the structure of convolutional and fully connected layers and used mean pooling layers instead of fully connected layers for classification and proposed the GoogleNet model, which greatly reduces the number of model parameters and has a top 5 error rate of only 6.7%. However, as the depth of the network keeps increasing, gradient disappearance or gradient explosion inevitably occurs. In this case, as the classification accuracy gradually reaches saturation, continuing to deepen the network will, on the contrary, cause the classification accuracy to decrease and the reason for this phenomenon is the gradient degradation problem in the training process. The proposed deep residual network model effectively solves the above problem. In the literature [10], a ResNet model is proposed, which has only 18% of the number of parameters of VGG. This model improves the discriminative ability of automatically learning features by superimposing residual items in the convolutional layers, and it is an extremely deep convolutional neural network model, which avoids the gradient disappearance and gradient explosion problems on the one hand and solves the gradient degradation problem better on the other hand, and now, the model has been extended to more than 1000 layers.

The literature [11] proposes an instance-weighted domain adaptation method to solve the crosslingual text classification problem by first aligning the feature spaces of the source and target domains into a common space and then adjusting the weights of some instances in the source domain by feature iteration for migration learning. The literature [12] proposes a metric migration learning framework, which is based on DCNN (deep convolutional neural network) and further motivates successful migration of knowledge across domains by learning the weights of samples and the distance between the source and target domains through model training. The literature [13] proposes a deep integration migration learning method that combines integration learning and DCNN to improve the generalization performance of migration learning by segmenting and reorganizing the source domain data, effectively filtering redundant data, and integrating classifiers. Mapping-based deep convolutional neural network migration learning refers to mapping the data from the source and target domains into a new feature space in which the data from both domains are equally distributed. The literature [14] proposes migration component analysis methods to determine low-dimensional embeddings of crossdomain data so that

matching of crossdomain data can be achieved accordingly. Subsequently, considering the combination of TCA methods with DCNN, the literature [15] proposes the DDC (deep domain confusion) approach to solve the domain adaptation problem. To eliminate domain differences, DDC employs the AlexNet model for source and target domain data, while a separate layer is introduced, and the MMD distance is added at layer 7 (the upper layer of softmax) to reduce the differences between the source and target domains. Subsequently, the literature [16] improved the previous work by introducing multicore MMD distances instead of MMD distances and proposed the DAN (deep adaptation networks) model to solve the domain adaptation problem. DAN maps the source and target domains into an RKHS (reproducing kernel Hilbert space) and then finds the mean difference between the mapped source domain data and the target domain data and then applies multilayer adaptation to the higher-level part of DCNN. After that, the literature [16] proposes a joint maximum mean difference method to measure the relationship of joint distributions, which is used to improve the generalization ability of DCNN to perform migration learning and thus adapt the data distribution between different domains. The literature [17] points out that deep neural network models are more powerful in feature learning and can be initialized layer by layer to alleviate the complexity of deep models in training, and this paper marks the beginning of a new wave of deep learning.

3. A Study on the Intelligent Translation Model for the English-Incorporating Neural Network Migration Learning Algorithm

3.1. Fusion of Neural Networks and Migration Learning Algorithms. The essence of transfer learning is the transfer of knowledge from other domains for reuse. Mathematically, transfer learning consists of two concepts, i.e., the domain and learning task. Based on the definition of transfer learning, transfer learning can be divided into three types according to the differences between source and target domains and source and target tasks, which are inductive transfer learning, direct push transfer learning, and unsupervised transfer learning. The rule-based approach is implemented by programming computer experts and hand-written translation rules by linguists. However, this method has significant drawbacks. The first is that the human-made rules cannot cover all utterances and can only correctly translate the sentences that satisfy the human-made rules, but not the sentences for which no rules have been made before. By summarizing the migration learning methods in recent years, they can be classified into four major categories.

- (1) Instance-based migration learning. The main principle of this method is to select the data in the source domain by weighting them and then increase the weights of the samples that are similar to the target domain data and decrease the weights of the data that are not similar to the target domain data. This increases the degree of similarity between the source

domain data and the target domain data, thus effectively facilitating the knowledge migration from the source domain to the target domain

- (2) Feature-based migration learning. This method reduces the differences between domains by finding an effective common feature representation in the source and target domain feature space and achieves migration of the feature representation between domains, which in turn improves the learning performance of the target domain
- (3) Transfer learning based on associative knowledge. This method refers to the discovery of associative knowledge such as rules, structures, and logics shared among similar domains and migration by establishing suitable mappings to improve the learning performance of the target domain by migrating the associativity between data
- (4) Parameter-based migration learning. This method mines model parameters or prior knowledge shared between models in the source and target domains and then establishes a link between the target and source tasks
- (5) Model-based migration learning. This method is usually combined with deep learning models to migrate the structure and model parameters of models already trained on large-scale datasets (e.g., AlexNet, VGGNet, and ResNet) to a new task and distinguish different levels of feature migration ability to fine tune the training process of the model, which is widely used in the field of language recognition and provides a new idea for recognition of small-scale datasets

The study of feature representation of language has been a very popular research topic in the field of language recognition. For recognition tasks, the degree of goodness of the feature representation usually directly determines the result of the final algorithm. A good feature extraction algorithm can not only extract the key information with discriminative features but also help us to better recognize and understand the whole language. Therefore, a good feature representation generally has the following two characteristics [18].

- (1) Discriminative performance. A good feature representation algorithm should be able to capture the most direct difference features between different patterns and be able to aggregate the fragmented distinguishing information between different categories to improve discriminatory performance
- (2) Generalization performance. Generalization performance is a basic requirement for all application algorithms, and the language feature representation algorithm should have good generalization performance in addition to being able to extract discriminative features from the language. If linguistic feature representation can only be applied to a specific

FIGURE 1: Model of the linguistic feature representation method.

problem in a specific domain, its applicability in real life will be greatly reduced. The detection operators of traditional feature extraction methods are generally artificially designed and obtained after a large amount of a priori knowledge summarization. For example, SIFT and HOG operators have the best performance over linguistic feature representation 10 years ago. Due to the variability and complexity of languages, machine translation performance still has a long way to go to achieve the desired goals. However, since extremely complex scenarios are usually encountered in practical applications, the discriminative performance and generalization performance of feature extraction operators that rely on human design and experience are also greatly limited. A model of the linguistic feature representation approach is shown in Figure 1

Now with deep convolutional neural networks (DCNN) making breakthroughs in the field of computer vision, features constructed using CNN's feature representation can advance the experimental results even further. Recent studies have shown that feature representations obtained by convolutional neural network learning have stronger discriminative performance compared to traditional manually designed features. DCNN is a self-learning feature representation method that can obtain better representation than SIFT and HOG empirically designed feature extraction operators. In addition, a huge amount of research results show that the models obtained from deep learning training have strong migration ability. The feature representation obtained by deep learning proves to have excellent generalization performance, and the improvement of DCNN recognition rate does not need to be based on a large number of training

samples and can directly migrate the pretrained model, so it can further extend the application scope of deep learning [19]. In our fine-grained image recognition application, I introduce a DCNN-based migration learning approach to obtain better recognition results and generalization capability than manually designed features.

Data preprocessing is an important process in model training, which requires adjusting the data to a standard normal distribution. However, in the process of training deep convolutional neural networks, the data needs to pass through multiple layers of networks and the weights and biases affect the accuracy of the final output. To alleviate the above phenomenon, a batch normalization (BN) method is used to deal with the problems associated with internal covariance shifts in feature mapping through a batch normalization (BN) method. The internal covariance shift is a change in the distribution of hidden unit values that forces the learning rate to a minimum, slows down the convergence of the model training process, and requires careful initialization of the parameters. The batch normalization of the transformed feature mapping x is shown in the following equation.

$$\vec{x} = (x_1, x_2, x_3, x_4, \dots, x_n). \quad (1)$$

To prevent model overfitting, I add a weight decay term as a regularization term to the loss function, where n denotes the number of data in the training set, y denotes the regularization coefficient, and w is the weight parameter of the network:

$$g_{i,j,f}^1 = \text{ReLU} \left(b_1 + \sum_{a=1}^m \sum_{b=1}^n g_{i,j,f} + b \cdot t_{a,b,c}^1 \right). \quad (2)$$

FIGURE 2: Deep convolutional neural network structure with fused migration learning.

The correct prediction rate was chosen as the model performance metric, and the correctness (accuracy) was defined as

$$\prod_i p(\otimes | > i) = \prod_i p(> i \otimes). \quad (3)$$

Considering that the acquired signal is usually contaminated by noise, larger convolutional kernels are used in the convolutional layer; large convolutional kernels provide a larger field of perception and can better suppress high-frequency noise compared to small-sized convolutional kernels. Smaller multiscale convolutional kernels are used in the later convolutional layers, which can better extract and distinguish features from different classes of data and increase the feature representation capability of the network:

$$R^A(t) = \frac{\partial \left(\left([\alpha_a^2(t)]^k [\beta_b^2(t)]^y \right) / \left(\sum_{m \neq a, n \neq b} [\alpha_m^2(t)]^k [\beta_n^2(t)]^y \right) \right)}{\partial t \cdot \partial \gamma}. \quad (4)$$

When training with an algorithmic model, our goal is to measure the difference between the true and predicted values and the first half of equation (2) will have no impact on our target task. So in machine learning tasks, it is usually straightforward to choose crossentropy as our loss function.

$$r(h_i) = Nh_i(1) \cdot (Nh_i(1) + Nh_i(-1))^{-1}. \quad (5)$$

Batch normalization enables faster training by using a higher learning rate and alleviates the problem of poor initialization. BN can also use saturated nonlinearity by preventing the network from falling into saturated patterns. In summary, batch normalization is a distinguishable transformation that introduces normalized activation into the network. In practice, the BN layer can be inserted immediately after the fully connected layer [20].

DCNN is a multilayer feedforward neural network that uses a set of convolutional kernel pairs for multiple transformations at each layer. Convolutional operations help to extract useful features from locally relevant data by assigning the output of the convolutional kernels to nonlinear processing units, and this nonlinearity produces different activation patterns for different responses, thus helping to learn semantic differences in images. DCNNs are specifically designed to process images so that the neurons in each layer are organized in three dimensions—height, width, and depth—just as pixels in an image will distinguish between different color values. A CNN with automatic feature extraction reduces the need for separate feature extractors. The important properties of DCNN are hierarchical learning, automatic feature extraction, multitasking, and weight sharing. It is mainly composed of convolutional, excitation, pooling, and fully connected layers. Deep convolutional neural network-based migration learning is introduced in fine-grained language recognition tasks to solve the fine-grained text data volume, the problem of an insufficient amount of fine-grained text data. For DCNN models, migration learning uses the knowledge gained from training on large datasets and then migrates to new domains to help the learning of new tasks, which can effectively solve the problem of insufficient training samples in specific domains. Figure 2 shows the classical DCNN structure applied to language recognition. A DCNN model is first trained on an ImageNet large-scale image dataset to determine the network weights and give the network feature representation capabilities. This knowledge is then migrated to the fine-grained image recognition task, using these pretrained networks as feature extractors to obtain feature representations of the fine-grained image dataset.

3.2. A Study of Intelligent Translation Models for the English-Incorporating Neural Network Migration Learning Algorithms. English is a low-resource language and there is not a large amount of English corpus, especially in some specific domains. Therefore, training a neural network with a small amount of English corpus will not work very well.

The proposed solution in this study is to train an English-Chinese neural machine translation model first and then use the migration learning technique to migrate to the original model and finally get an English-Chinese neural machine translation model.

When the sentence we need to translate is particularly long, an information vector C may not fully represent all the information of our source input sentence, thus causing a loss of information; one solution is to extract all the hidden outputs of the RNN encoding process into our decoder, but this would be huge and unfocused, so an attention mechanism needs to be introduced to extract all the encoded information that is extracted. Figure 3 shows the encoder-decoder framework with the attention mechanism.

The many to many structures are used in machine translation, and this structure is also used for tasks such as chat conversations. There are many applications of the encoder-decoder framework. The framework generally refers to encoding one input into one message and then decoding the message to get one input; this framework is generally used in image classification scenarios. One, the length of the context is precisely controlled by the superposition of the convolution; two, the convolution allows parallel computation, which is not dependent on the previous moment state and allows parallelism on each element in the sequence; three, the training complexity is reduced and the number of convolution kernels and nonlinear computations that pass through each word of the input convolutional network is fixed. Many frameworks generally refer to encoding one input to produce multiple outputs; this framework is generally used in inputting one image to produce multiple descriptions of this image. The many-to-one framework generally refers to taking many inputs and ending up with only one output, and this framework is generally applied to text sentiment classification where multiple words of a sentence are input to output the sentiment of that sentence. Another framework for many to many is the synchronous many-to-many framework, whose typical structure is that each moment of input has a corresponding output and the output of this moment is fed to the next moment as input, which can be used for character prediction, video tagging, video classification, etc.

Neural network machine translation systems require a large amount of parallel corpus for training the model, and the data resources are extremely unbalanced for different small languages and dialects; large languages can obtain enough parallel corpus for neural network model training due to their wide application or a large population of speakers, while for most small languages and dialects, the available parallel corpus data resources are extremely scarce. In the field of machine translation, there are usually two cases where migration learning can be used: one is when the target domain has a small amount of data, and migration learning can be performed by tasks with sufficient data and similar to that task (source domain); the other is when the model is complex and retraining the model is time-consuming, and learning efficiency can be accelerated by migration learning. The idea of migration learning domain adaptation is that for some specific domains when sufficient data is not

FIGURE 3: Encoder-decoder framework with the introduction of attention mechanism.

available, the trained models from other domains can be migrated and then fine tuned for that model. For a target language with low resources, domain-adaptive transfer learning can be performed on a model trained on another language with more abundant resources using a small amount of adaptive data. However, this domain-adaptive migration learning approach tends to lead to overfitting problems when training neural network machine translation models and it is difficult to converge during training, especially when the model parameters are too large. This is because the system usually works best when the ability of the neural network machine translation model to fit various functions matches the complexity of the translation task being performed and the amount of training data provided. As usual, models with high capacity can solve complex tasks but when their capacity is higher than required by the task, overfitting problems arise and affect the generalization ability of the end-to-end neural network machine translation model. The central issue in the design of end-to-end neural network machine translation models is not only to perform well on training data but also to be able to generalize strongly to new translation tasks, including low-resource target languages.

In a real machine translation task, the performance of a translation model that we train with a parallel corpus dataset of one language for another language will be significantly degraded, because the distribution of the test and training sets will be very different. Domain adaptation is a type of migration learning that improves the performance of a target domain model by using data samples from an information-rich source domain. The domain adaptation problem consists of two parts, the source domain, and the target domain. The source domain is generally rich in supervised (labeled) information, while the target domain has only a small amount of labeled information or there is no labeled information. Domain adaption includes three methods: (1) sample domain-adaptive methods, by weighted

resampling of samples in the source domain and then approximating the target domain distribution; (2) feature-level domain-adaptive flip methods, by extracting common features in the source and target domains; and (3) model-level domain-adaptive methods, by adjusting the errors in the source and target domains. Model-level domain adaptation uses model migration techniques to migrate models trained in the source domain using large amounts of data to be applied to the target domain. For example, in the task of machine translation, when a new language translation problem is encountered, the original model trained in the source domain is simply migrated to the new target domain, so that the new translation task requires less parallel corpus to be able to train a better machine translation model and the machine translation system can get high performance under low-resource conditions. However, the mismatch between models and data is prone to overfitting problems, which further affects the generalization ability of the translation system.

4. Experimental Verification and Conclusions

In this chapter, we described in detail the source, acquisition, postacquisition processing, and cleaning of the Indian English-Chinese bilingual corpus, so what we do below is to label each word of the sentences in the corpus with lexicality and then use the corpus with lexicality labeling to fine tune. The English-Chinese machine translation model has already been trained. We use the LTP tool to annotate our corpus, and we choose the LTP tool here because it has an accurate lexical annotation algorithm and it is also a common lexical annotation tool used by many research institutions and enterprises.

When there is a difference between the source domain data and the target domain data, the larger the difference, the worse the migration effect and even a negative migration effect when the difference between the data is too large, i.e., the source domain data not only cannot assist the target domain model learning but even interfere with the training of the target domain model and reduce the model performance. However, at the same time, text convolution and SAR convolution still have some similarities in the underlying structural features. From the visualization results of AlexNet and Mnet2 convolution kernels in Figure 4, we can see that the convolution kernels involving structural features in the textual convolution kernel and SAR convolution kernel are the same in shape and the similarity between the underlying features lays the foundation for model migration. The alignment matrix of the source and target sequences shows the important distribution of each word of the source sequence for the current word to be translated when a word is translated. To verify the effectiveness of migrating the optical model to the SAR image domain, the t-SNE algorithm is used to visualize the distribution of the original MSTAR data, and then, the MSTAR 10-class data is input to a fully trained VGG-16 model on the ImageNet dataset and the FC6 fully connected layer output of the VGG-16 model is extracted as the high-dimensional features and then

FIGURE 4: Comparison of original data and extracted feature distribution.

the t-SNE algorithm to visualize the distribution of these high-dimensional features.

Overfitting refers to the process of fitting the model parameters because the training data contains sampling error and the model fitting ability is too strong, resulting in the model fitting the sampling error of the training data as well. The specific performance is that the model has a good classification effect on the training set but a very poor effect on the test set and the model generalization ability is weak. In the training process of migration learning neural networks, if overfitting occurs, it usually shows as the training loss Train_loss converges while the test accuracy Test_accuracy rises and then falls, as shown in Figure 5. The common methods to solve overfitting are model lightweight, early stopping, regularization, and data expansion. Early stopping is to terminate the training before overfitting occurs to prevent the model accuracy from degrading; L1 regularization can make the weight matrix more sparse and L2 regularization can control the weights to become smaller. Both regularization methods can reduce the complexity of the network to alleviate overfitting, while data expansion is used to reduce the sampling error by giving the model more training data, thus alleviating overfitting. In addition to these common methods, network pruning can also alleviate the overfitting problem. Network pruning is an effective method to reduce network complexity and overfitting in early convolutional neural network research, which can enhance the generalization performance of the network by removing a large number of redundant parameters while keeping the accuracy of the model largely unchanged.

In the heterogeneous network model migration framework, when training a two-stream CNN network, we need to extract the features of a layer of the source and target

FIGURE 5: Example of overfitting of a migration learning neural network model.

branches as the output of the two branches and then constrain the parameter update process of the target branch model by measuring the distance between the output features of the two branches with the help of the loss function in reverse. We have two options for the selection of the feature constraint layer, i.e., choosing the last fully connected layer as the feature constraint layer or the output of softmax as the feature constraint layer. In subsequent experiments, we refer to the approach of using the output of the softmax layer as the feature constraint layer as framework 1 and the approach of using the output of the last fully connected layer as the feature constraint layer as framework 2. The model shear rate is a hyperparameter, and there is no literature on which model shear rate is optimal, but fully connected layers can usually be set at a high model shear rate due to a large number of parameters and numerous redundant parameters. The convolutional layer, on the other hand, has a small number of parameters, and a slightly higher parameter shear rate can cause a significant drop in model accuracy. Figure 6 shows the experiments of parameter shear rate versus model classification accuracy for the fully connected layer and the convolutional layer, and it can be seen that the loss of model classification accuracy starts when the parameter shear rate of the convolutional layer exceeds 25%, while the fully connected layer can withstand a higher parameter shear rate with no change in model accuracy.

The model pruning operation is performed before the migration of the dual-stream CNN target domain branching model to the target domain, and its purpose is to alleviate model overfitting and further enhance the migration effect. This operation is independent of other steps within the method and does not affect each other, whereby I divide the above factors into two groups to conduct the trials, that is, heterogeneous network model migration experiments with the model pruning operation and experiments without the model pruning operation. I use the MSTAR10 class 22 data as the experimental data, and first, we give the model classification accuracy of VGG-16 and Mnet10 without expanding the training data and without migration operation. Compared with the VGG-16 model, the classification

FIGURE 6: Sensitivity of convolutional and fully connected layers to parameter shear.

FIGURE 7: Demonstration of the model training process in the migration experiment.

accuracy of Mnet10, which is designed for SAR images, is nearly 10 percentage points higher than that of VGG-16 in the MSTAR decile task because of its lightweight structure and good discrete feature extraction capability. For different tasks in the field of natural language processing, different corpora are used for word vectors. The word vector model has been studied in depth for different tasks in natural language processing using different corpora. This is the reason why I do not directly employ the VGG-16 model in the SAR image domain. The results in Figure 7 will be used as the baseline accuracy of the Mnet10 model and will be compared with the results of the two subsequent sets of experiments as a way to test the effectiveness of the heterogeneous network model migration approach. It can be seen that in the eight sets of experiments, the model has largely

FIGURE 8: Experimental results of the model shear rate.

converged by about 3000–4000 iterations when the training loss is small and no longer fluctuates. However, when the model converges, the accuracy starts to drop significantly when the model continues to be trained for 10000 iterations, which means that the model has a serious overfitting problem, so I adopt the early stopping strategy in this experiment, that is, I terminate the training early when overfitting occurs.

I again conducted experiments on the selection of the model shear rate in model migration between heterogeneous networks. In this set of experiments, I use the L1-SOFT-P approach for migration, i.e., I use the softmax layer as the feature constraint layer and the L1 parametrization as the metric to train the dual-stream CNN and then migrate only the parameters of the feature extraction part of the target-domain branch model. Before migrating the target domain branch model to the target domain, I added a model pruning operation to prune the target domain branch model according to different model pruning rates and then migrated some parameters of the target domain branch model to obtain the final experimental results. Since there are 12 experimental scenarios, there are too many lines in Figure 8, so to show the experimental results more clearly, I show the scenarios where the migration effect decreases after model pruning with short dashed lines, the scenarios where the migration effect remains more or less the same after model pruning with long dashed lines, and the scenarios where the migration effect improves after model pruning with solid lines. The model pruning operation does improve the effect of heterogeneous network model migration by up to about 3 percentage points. However, as can be seen in Figure 8, two experimental scenarios outperform the results without model pruning at 4000 to 10000 training iterations. The reason for this phenomenon may be that the existing model pruning methods have too simple criteria for discriminating the importance of parameters within the model and we did not perform iterative pruning operations during the pruning procedure, resulting in some important parameters being

lost during the migration process, which in turn makes most of the pruning experimental scenarios unsatisfactory. Again in the model overfitting section in Figure 8, it can be seen that the results of almost all experimental scenarios are better than the results without model pruning. This demonstrates the effectiveness of the model pruning approach in mitigating the overfitting problem. Thus, the model pruning operation can alleviate the overfitting problem in heterogeneous network model migration, can improve the final model migration, and may have higher improvement after replacing the model with a new model pruning method.

Two heterogeneous networks in the source and target domains are combined into a two-stream CNN, and the loss function is used to constrain the two branches to produce approximate outputs under the same input condition during training, which makes the mapping function of the target domain branch model gradually converge to that of the source domain branch model and finally achieves the model migration between the heterogeneous networks. At last, to further improve the migration effect, the model pruning operation is also introduced in the framework to solve the overfitting problem in the migration process. The experimental results based on MSTAR 10 class 22-type data demonstrate the effectiveness of the model migration method among heterogeneous networks.

5. Conclusion

The research work in this paper focuses on English-Chinese neural machine translation research; from bilingual corpus, language features, neural networks, migration learning, and other aspects are introduced. English is affected by many factors and has many unique linguistic features, and the number size is not very large—the study of building English-Chinese bilingual corpus, the study of English-Chinese neural machine translation, the study of migration learning, and the study of incorporating linguistic features in neural networks. Firstly, this paper makes a relevant study

on the unique linguistic features that English has, mainly lexical features and syntactic features, and it is necessary to study the lexical features and syntactic features of English to be able to improve the performance of neural machine translation models. In the absence of a massively parallel corpus, the machine translation task can be converted into an unsupervised task by first using a massively monolingual corpus for language model pretraining to obtain a whole. The unsupervised machine translation technique is a good way to improve machine translation under low-resource conditions. The unsupervised machine translation technique is a good solution to improve the performance of machine translation under low-resource conditions. Our ultimate goal is to train an English-Chinese neural machine translation model, but since the English corpus is relatively small, using a separate neural network approach to do the training may end up with poor results, so to solve the problem of the small amount of training corpus, I use the idea of migration learning and use the fine tune technique to use a small amount of corpus to retrain the pretrained model, so that the model performance of the trained neural network will be much better.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The author does not have any possible conflicts of interest.

References

- [1] X. Wang, C. Chen, and Z. Xing, "Domain-specific machine translation with recurrent neural network for software localization," *Empirical Software Engineering*, vol. 24, no. 6, pp. 3514–3545, 2019.
- [2] M. Alam, M. D. Samad, L. Vidyaratne, A. Glandon, and K. M. Iftekharruddin, "Survey on deep neural networks in speech and vision systems," *Neurocomputing*, vol. 417, pp. 302–321, 2020.
- [3] T. Chen, S. Liu, Y. Li et al., "Developed and validated a prognostic nomogram for recurrence-free survival after complete surgical resection of local primary gastrointestinal stromal tumors based on deep learning," *eBioMedicine*, vol. 39, pp. 272–279, 2019.
- [4] J. Zheng, Y. Chen, C. Wu, X. Shi, and S. M. Kamal, "Enhancing neural sign language translation by highlighting the facial expression information," *Neurocomputing*, vol. 464, pp. 462–472, 2021.
- [5] I. Chalkidis and D. Kampas, "Deep learning in law: early adaptation and legal word embeddings trained on large corpora," *Artificial Intelligence and Law*, vol. 27, no. 2, pp. 171–198, 2019.
- [6] Q. D. Nguyen, D. A. le, N. M. Phan, and I. Zelinka, "OCR error correction using correction patterns and self-organizing migrating algorithm," *Pattern Analysis and Applications*, vol. 24, no. 2, pp. 701–721, 2021.
- [7] M. AlSallal, R. Iqbal, V. Palade, S. Amin, and V. Chang, "An integrated approach for intrinsic plagiarism detection," *Future Generation Computer Systems*, vol. 96, pp. 700–712, 2019.
- [8] S. Pirasteh, G. Shamsipour, G. Liu, Q. Zhu, and Y. E. Chengming, "A new algorithm for landslide geometric and deformation analysis supported by digital elevation models," *Earth Science Informatics*, vol. 13, no. 2, pp. 361–375, 2020.
- [9] H. J. Park, J. Y. Jang, and J. H. Lee, "Assessment of rainfall-induced landslide susceptibility at the regional scale using a physically based model and fuzzy-based Monte Carlo simulation," *Landslides*, vol. 16, no. 4, pp. 695–713, 2019.
- [10] S. Singh, V. Sood, R. Kaur, and S. Prashar, "An efficient algorithm for detection of seasonal snow cover variations over undulating North Indian Himalayas, India," *Advances in Space Research*, vol. 64, no. 2, pp. 314–327, 2019.
- [11] M. Egger, M. Ley, and S. Hanke, "Emotion recognition from physiological signal analysis: a review," *Electronic Notes in Theoretical Computer Science*, vol. 343, pp. 35–55, 2019.
- [12] R. L. Soteres, J. Pedraza, and R. M. Carrasco, "Snow avalanche susceptibility of the Circo de Gredos (Iberian Central System, Spain)," *Journal of Maps*, vol. 16, no. 2, pp. 155–165, 2020.
- [13] G. Wilson, C. Pereyda, N. Raghunath et al., "Robot-enabled support of daily activities in smart home environments," *Cognitive Systems Research*, vol. 54, pp. 258–272, 2019.
- [14] Y. Zheng, Q. Song, J. Liu, Q. Song, and Q. Yue, "Research on motion pattern recognition of exoskeleton robot based on multimodal machine learning model," *Neural Computing and Applications*, vol. 32, no. 7, pp. 1869–1877, 2020.
- [15] W. Liu, M. Zhang, Y. Zhang et al., "Real-time multilead convolutional neural network for myocardial infarction detection," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 5, pp. 1434–1444, 2018.
- [16] M. Ebrahimi, H. Nejadsoleymani, and M. R. Mansouri Daneshvar, "Land suitability map and ecological carrying capacity for the recognition of touristic zones in the Kalat region, Iran: a multi-criteria analysis based on AHP and GIS," *Asia-Pacific Journal of Regional Science*, vol. 3, no. 3, pp. 697–718, 2019.
- [17] L. Xiao, Y. Zhang, T. Ge, C. Wang, and M. Wei, "Analysis, assessment and early warning of mudflow disasters along the Shigatse section of the China–Nepal highway," *Open Geosciences*, vol. 12, no. 1, pp. 44–58, 2020.
- [18] H. Li, S. Qi, X. Yang, X. W. Li, and J. W. Zhou, "Geological survey and unstable rock block movement monitoring of a post-earthquake high rock slope using terrestrial laser scanning," *Rock Mechanics and Rock Engineering*, vol. 53, no. 10, pp. 4523–4537, 2020.
- [19] Y. Wang, S. Cang, and H. Yu, "A survey on wearable sensor modality centred human activity recognition in health care," *Expert Systems with Applications*, vol. 137, pp. 167–190, 2019.
- [20] I. Fathi, Q. I. Ali, and M. Abdul-Jabbar, "Real-time voice transmission over wireless sensor network (VoWSN) based automatic speech recognition (ASR) technique," *Al-Rafidain Engineering Journal (AREJ)*, vol. 24, no. 2, pp. 23–34, 2019.