

Article

Prompt Tuning for Multi-Label Text Classification: How to Link Exercises to Knowledge Concepts?

Litong Wei ^{1,2} , Yun Li ^{1,*}, Yi Zhu ¹ , Bin Li ¹ and Lejun Zhang ¹ ¹ School of Information Engineering, Yangzhou University, Yangzhou 225012, China² Jiangsu Provincial Key Constructive Laboratory for Big Data of Psychology and Cognitive Science, Yancheng Teachers University, Yancheng 224002, China

* Correspondence: liyun@yzu.edu.cn; Tel.: +86-13852785198

Abstract: Exercises refer to the evaluation metric of whether students have mastered specific knowledge concepts. Linking exercises to knowledge concepts is an important foundation in multiple disciplines such as intelligent education, which represents the multi-label text classification problem in essence. However, most existing methods do not take the automatic linking of exercises to knowledge concepts into consideration. In addition, most of the widely used approaches in multi-label text classification require large amounts of training data for model optimization, which is usually time-consuming and labour-intensive in real-world scenarios. To address these problems, we propose a prompt tuning method for multi-label text classification, which can address the problem of the number of labelled exercises being small due to the lack of specialized expertise. Specifically, the relevance scores of exercise content and knowledge concepts are learned by a prompt tuning model with a unified template, and then the multiple associated knowledge concepts are selected with a threshold. An Exercises–Concepts dataset of the Data Structure course is constructed to verify the effectiveness of our proposed method. Extensive experimental results confirm our proposed method outperforms other state-of-the-art baselines by up to 35.53% and 41.78% in Micro and Macro F1, respectively.

Keywords: linking exercises to concepts; multi-label text classification; prompt tuning; few-shot



Citation: Wei, L.; Li, Y.; Zhu, Y.; Li, B.; Zhang, L. Prompt Tuning for Multi-Label Text Classification: How to Link Exercises to Knowledge Concepts? *Appl. Sci.* **2022**, *12*, 10363. <https://doi.org/10.3390/app122010363>

Academic Editors: Jing Zhang, Jipeng Qiang and Cangqi Zhou

Received: 21 September 2022

Accepted: 10 October 2022

Published: 14 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent decades, personalized learning has become a mainstream solution to enhance students' learning interest, and experience in intelligent education systems [1–3]. One of the fundamental and key tasks in personalized learning is knowledge tracing [4,5], which aims to evaluate the students' learning state of knowledge concepts.

Exercises have played an important role in the knowledge tracing tasks, which is one of the evaluation metrics of whether students have mastered specific knowledge concepts [6,7]. Students in intelligent education systems choose the right exercises according to their own needs and acquire specific knowledge concepts during exercise. In turn, we can track changes in students' acquisition of knowledge concepts during their exercising process. From this perspective, knowledge tracing should consist of a students–exercises–knowledge concepts hierarchy [8]. However, most existing methods of knowledge tracing approaches [9–11] are partially modeled among the hierarchy (i.e., students–exercises or students–concepts). This is because, in some intelligent systems, there is a lack of connection between exercises and knowledge concepts. To this end, we take the automatic linking of exercises to knowledge concepts into consideration for knowledge tracing tasks.

In essence, linking exercises to knowledge concepts is a multi-label text classification (MLTC) problem. As shown in Figure 1, the relationship between exercises and knowledge concepts is one-to-one or one-to-many, which aims to assign one or more concepts to each input exercise in the dataset. Moreover, Figure 1 shows that the semantics between exercises and knowledge concepts are highly correlated.

Exercises	labels
During the execution of the program, the recursive call to the implementation function should rely on the stack structure. (right)	stack
The stack is characterized by first in, last out, and the queue is characterized by first in, first out. (right)	stack queue
Sparse matrix means (C) A matrix with few elements B matrix with few zero elements C Matrix with few nonzero elements D Matrix with few rows and columns	array
The main advantage of circular linked list is (every node can be found backwards along any node).	Linked list
In the stack structure, the end that allows insertion is called (the top of the stack); in the queue structure, the end that allows insertion is called (the tail of the queue).	stack queue

Figure 1. Examples of exercises linking to knowledge concepts from dataset.

Recently, deep-learning-based methods have achieved fairly good performance in MLTC for the superiority of feature representation learning. For example, Liu et al. [12] utilized the strengths of the existing convolutional neural network and took multi-label co-occurrence patterns into account in the optimization objective to produce good results in MLTC. Pal et al. [13] proposed a graph attention network-based model to capture the attentive dependency structure among the labels. Chang et al. [14] fine-tuned the BERT language model [15] to capture the contextual relations between input text and the induced label clusters. However, these deep-learning-based methods in MLTC tasks require large amounts of training data for model optimization, which is usually time-consuming and labour-intensive in real-world scenarios. Unfortunately, linking exercises to knowledge concepts usually lacks training data because some knowledge concepts corresponding to a few exercises or new courses may contain a paucity of labelled data.

To address these problems, we propose a Prompt Tuning method for Multi-Label Text classification (PTMLTC for short). First, the prompt tuning model with a unified template predicts the relevance scores of exercises and knowledge concepts. Then, the multiple associated knowledge concepts are picked with a threshold. In order to verify the effectiveness of our proposed method, an Exercises–Concepts dataset of the Data Structure course is constructed. Extensive experimental results confirm our method outperforms other state-of-the-art methods by up to 32.53% and 41.78% in Micro and Macro F1, respectively.

The contribution of our paper can be summarized as follows:

(1) To the best of our knowledge, this is the first attempt to automatically link exercises to knowledge concepts. We built an Exercises–Concepts dataset of the Data Structure course and reconstructed the few-shot dataset.

(2) We propose a prompt tuning method for multi-label text classification to link exercises to knowledge concepts. Large amounts of labelled or unlabeled training data are not required.

(3) Extensive experimental results confirm that our proposed method outperforms other state-of-the-art deep-learning-based methods.

2. Related Work

In this section, firstly, we introduce the deep-learning-based multi-label text classification methods. Then, the prompt tuning learning methods used in our models will be presented.

2.1. Multi-Label Text Classification

The goal of MLTC is to associate one or more relevant labels for each input text instance. The traditional MLTC methods include one-vs-all methods [16,17], tree-based methods [18,19] and embedding-based methods [20,21]. For example, Babbar et al. [16] proposed a distributed learning mechanism for MLTC, which can use doubly parallel training to reduce the expensive computational cost of one-vs-all methods. Prabhu et al. [22] presented a method called FastXML by optimizing an nDCG-based ranking loss function to further reduce expensive computational costs. Tagami [21] proposed a graph embedding method, which learns partition data points by the k-nearest neighbour graph (KNNG) and uses an approximate k-nearest neighbour to predict results by exploring KNNG in the embedding space.

In recent years, due to the powerful ability of feature representations learning [23,24], deep models have gained much attention and achieved superior performances over traditional methods. The focus of existing deep-learning-based methods on MLTC is learning-enhanced text representation for improving performance. For example, Liu et al. [12] utilized the strengths of the existing convolutional neural network (CNN) and dynamic pooling to model the text representation for MLTC. Xiao et al. [25] employed an attention mechanism to explore highlight important context representation in MLTC tasks. Ma et al. [26] utilized the bidirectional Gated Recurrent Unit network and hybrid embedding for learning the representation of the text level-by-level. Chang et al. [14] proposed to fine-tune the BERT language model [15] in order to capture the contextual relations between input text for MLTC.

In addition, recently, the dependencies or correlations among labels have demonstrated the ability to improve performance in most MLTC tasks. Along this line, many deep-learning-based methods have been proposed to model label dependencies. For example, Chen et al. [27] explored labels' correlations through Recurrent Neural Networks, which were used to predict labels one-by-one sequentially. Pal et al. [13] proposed a graph-attention network-based model to capture the attentive dependency structure among the labels. Yang et al. [28] treated MLTC tasks as a sequence generation problem and proposed a decoder structure to capture the dependencies between labels that selected the most informative words automatically while predicting different labels. Xun et al. [29] learned label correlation by introducing an extra CorNet module that is applied to a deep model at the prediction layer to enhance raw label predictions with correlation knowledge.

However, most existing deep-based MLTC methods require a large amount of labelled or unlabeled training data for model optimization, which is often time-consuming and labour-intensive. Therefore, designing methods that can achieve promising results in the few-shot scenario remain a huge challenge in real-world MLTC tasks.

2.2. Prompt Tuning

Prompt-based learning [30–32] is regarded as a new paradigm in natural language processing and has drawn great attention from multiple disciplines, which promotes the downstream tasks by using the pre-training knowledge as much as possible. Starting from the GPT-3 [33], Prompt tuning has demonstrated unique strengths in a variety of tasks, which contain text classification [32,34], relation extraction [35], event extraction [36] and so on. Prompt-based learning directly models the probability of text on top of language models. It is different from traditional supervised learning, which trains a model to predict the output y as $P(x | y)$ with the input x . Specifically, in the prediction task, firstly, a template is added to the original input x to form a new textual string prompt x' with [MASK]. Then, the reconstructed \hat{x} is learned with the language model to probabilistically

fill the unfilled information. For example, Cui et al. [37] employed closed prompts filled by a candidate named entity span as the target sequence in named entity recognition tasks. Li et al. [38] proposed Prefix-tuning that uses continuous templates to improve performance than discrete prompts. There has already been some recent effort in devoting external knowledge to prompt design. For example, Hu et al. [34] proposed a knowledgeable prompt-tuning by expanding the label word space of the verbalizer with external knowledge bases. Chen et al. [35] proposed a knowledge-aware prompt-tuning approach, which introduced relation labels knowledge into prompt construction. In addition, many works [34,39] have demonstrated that prompt-based learning greatly improves model performance in few-shot scenarios. Hambardzumyan et al. [40] proposed an automatic prompt generation method to transfer knowledge from large Pre-trained Language Models, which achieved excellent performance in a few-shot setting. Gu et al. [41] proposed to add soft prompts into the pre-training stage and pre-train soft prompts in the form of unified classification tasks, which can reach or even outperform in few-shot settings. However, in the knowledge tracing tasks, we are not aware of existing prompt-learning-based approaches that automatically link exercises to knowledge concepts. To this end, we propose a prompt tuning method for multi-label text classification to link exercises to knowledge concepts.

3. Prompt Tuning Method for Multi-Label Text Classification

In this section, the details of our proposed PTMLTC are given, and the general framework is shown in Figure 2.

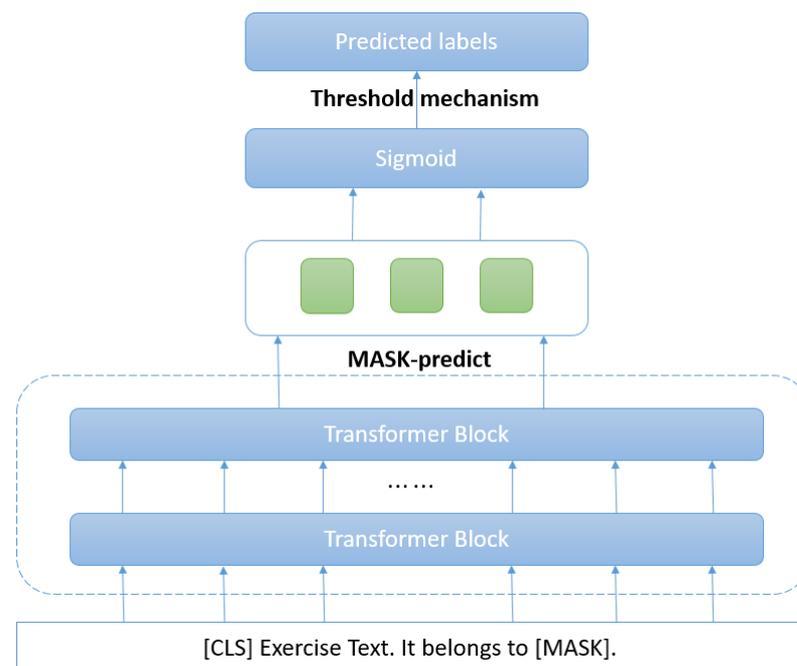


Figure 2. The general framework of our PTMLTC. Exercise Text sequence is connected with united template as the input of prefix Language Model. It will then predict the probability of filling the token [MASK] with each word of knowledge concepts. $Sigmoid()$ function is used to obtain the probability of exercise texts linking to knowledge concept labels. Finally, a threshold mechanism is adopted to predict all the possible knowledge concept labels.

3.1. Problem Formalization

In this paper, we aim to use few exercises with labeled concepts to predict one or more related concepts for each input exercise text. Given $C = \{c_1, c_2, \dots, c_N\}$ is the label space with N concepts, the goal is to learn a function $h(\cdot) : E \rightarrow 2^C$ from the support set $S = \{(E_i, C_i)\}_i^{N_s}$, where E denotes the exercise-instance space, S usually contains K exercise-instances (K -shot) of N concept-labels (N -way), N_s is the size of the support set.

For each learning instance (E_i, C_i) , $E_i \subseteq E$ is l -dimensional input and $C_i \subseteq C$ is the related concepts set. For an unseen instance e in the query set, the classifier predicts a set of concepts $P = h(e) \subseteq C$.

3.2. Prompt Tuning Method for Multi-Label Text Classification

As is shown in Figure 2, our methods adopt a threshold-based strategy [42,43] to achieve multi-label text classification. Firstly, the relevance scores of exercise content and knowledge concepts are transformed into a masked language model by prompt tuning methods. Specifically, a prompt template is defined as $V_{prompt} = "It\ belongs\ to\ [MASK]"$. and combine the exercise text $x = \{x_0, x_1, x_2, \dots, x_n\}$ to form the final input for prompt tuning input e_{prompt} , which can be shown as Equation (1):

$$e_{prompt} = [CLS]x, It\ belongs\ to\ [MASK]. \tag{1}$$

Suppose that M is a large corpus of Pre-trained Language Models (PLMs in short), the probability of filling the token [MASK] for each word of concept c in the knowledge concepts set C can be denoted as $P_M([MASK] = c | e_{prompt})$. Here, we need a map function $Sigmoid()$ to predict the probability of each concept independently. The relevance scores can be represented as (2):

$$P(c | e_{prompt}) = Sigmoid(P_M([MASK] = c | e_{prompt})) \tag{2}$$

Finally, we add an additional threshold mechanism to determine knowledge concepts corresponding to exercises, which can be formulated as (3):

$$P(e) = \{c | P(c | e_{prompt}) > t, c \in C\} \tag{3}$$

where t is the threshold.

To better introduce our method, we take an example shown in Figure 3. The exercise text "The stack is characterized by first in, last out, and the queue is characterized by first in, first out. (right)" is wrapped with template as the input. PLM is adopted to predict the predict the probability of filling the token [MASK] with knowledge concepts word set array, stack, queue, linked list. Then, $Sigmoid()$ function is used to obtain the probability of exercise text linking to labels $\{array, stack, queue, linkedlist\}$. Due to the probability of exercise text linking to stack, queue greater than threshold, exercise text is regarded as linking to $\{stack, queue\}$.

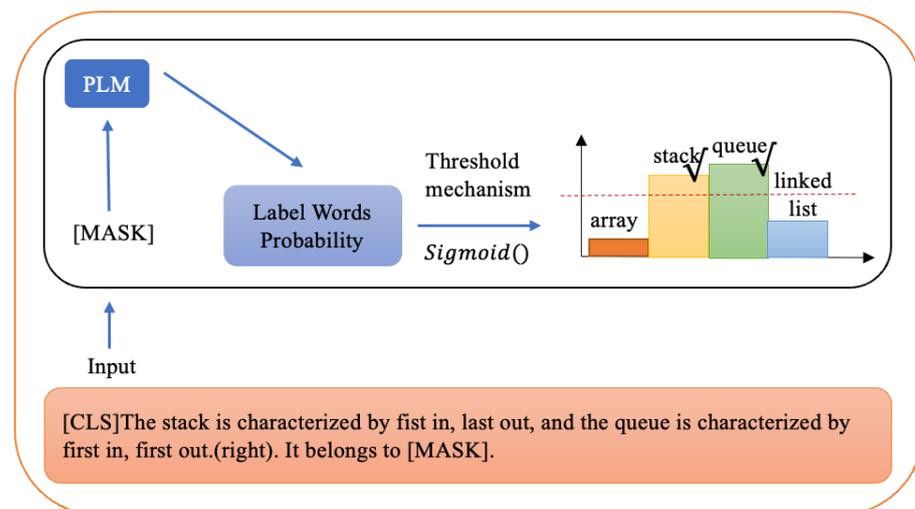


Figure 3. An example of our proposed method.

It has been proven that binary cross-entropy loss (BCE) over sigmoid activation is more suited for multi-label problems and outperforms cross-entropy loss [12]. Therefore, in our paper, the BCE loss function is chosen to learn parameters in the tasks, which can be formulated as (4):

$$\min_{\Theta} -\frac{1}{K} \sum_{i=1}^K \sum_{j=1}^N [y_{ij} \log(\sigma(\hat{p}_{ij})) + (1 - y_{ij}) \log(1 - \sigma(\hat{p}_{ij}))] \quad (4)$$

where \hat{p}_{ij} represents the predicted value of exercise i belongs to concept j . y_{ij} represents the value of exercise i belongs to concept j , and σ is the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$.

4. Experiment

In this section, we conduct extensive experiments on the constructed Exercises–Concepts dataset of the Data Structure course to verify the effectiveness of our proposed method for linking exercises to knowledge concepts. In the following, firstly, the Exercises–Concepts dataset of the Data Structure course and the few-shot dataset construction are introduced in detail. Then the compared methods and evaluation metrics of our experiments are shown. Finally, we analyze the experimental results and the influence of the main parameters.

4.1. Datasets

Exercises–Concepts dataset of Data Structure course: To study the problem of linking exercises to knowledge concepts, we construct the Exercises–Concepts dataset of the Data Structure course. Refer to MOOCube_DS [44] data repository and Several national planning textbooks, we extract 65 classic knowledge concepts. Subsequently, 2027 exercises used in these textbooks are marked with the corresponding knowledge concepts. Details are shown in Table 1.

Few-shot dataset construction: To simulate the few-shot situation, we reconstruct the dataset in to the form of few-shot learning, where each example is the combination of a query instance (e^q, c^q) and the corresponding $K - shot$ support set S . Unlike the single-label classification problem, instances of multi-label classification may be associated with multiple labels. Therefore, there is no guarantee that each label appears exactly K times during sampling. To address the problem, we approximately construct $K - shot$ support set S with the Minimum-including Algorithm [43]. It constructs a support set generally complying with the following two conditions: (1) All labels in the original dataset appear at least K times in support set S . (2) At least one label will appear less than K times in S if any (e^q, c^q) pair is removed from it. For the original dataset, we sampled N_S different support sets. For each support set, we take the remaining data as the query set. Each support-query-set pair constitutes one few-shot episode.

On the test stage, we constructed 10 different few-shot episodes for each selected K -shot. Among them, support set is used to fine tuning model, and query set is used to test the effectiveness of methods.

Table 1. The number of each exercise linking to knowledge concepts. *C* represents the label of the knowledge concept, *N* denotes the number of exercise.

<i>C</i>	<i>N</i>	<i>C</i>	<i>N</i>	<i>C</i>	<i>N</i>	<i>C</i>	<i>N</i>
Array	36	Sequential List	46	Bubble Sort	41	Binary Search	48
Logical Structure	48	Linked Storage Structure	48	Time Complexity	91	Generalized table	35
Heapsort	46	Physical Structure	35	Linked List	10	Matrix	35
Adjacency Matrix	35	Linear List	48	BT-Preorder Traversal	37	Tree-Degree	40
Algorithm	56	String	35	BT-Postorder Traversal	35	Tree-Depth	45
Queue	35	Tree	24	Minimum Spanning Tree	53	Graph	18
Recursion	39	Binary Tree (BT)	23	Topological Sort	45	Circular Queue	35
Complete Binary Tree	42	Binary Sort Tree	35	Depth First Search	5	Binary Tree-Threaded BinaryTree	35
Balanced BinaryTree	50	Huffman Tree	35	Breadth First Search	48	Shell's Sort	60
Search	136	Data Structure	35	Connected Graph	47	Binary Tree-Inorder Traversal	13
Sequential Search	53	Sequential Storage Structure	35	Quick Sort	68	Merge Sort	35
Critical Path	60	Stack	35	Full Binary Tree	48	Space Complexity	35
Selection Sort	35	Strongly Connected Graph	57	Graph-Degree	35	Selection Sort	35
HashSearch	35	Muitl-way Search Tree	35	Adjacency List	37	Sort	28
Shortest Path	35	Binary Tree-Order Traversal	56	Doubly Linked List	35	Straight Insertion Sort	44
Cycle Chain	14	Undirected Graph	27	Oriented graph	36	Data	12
Double Circle List	8						

4.2. Baselines and Evaluation

4.2.1. Baselines

Traditional deep-learning-based multi-label text classification methods, such as XML-CNN [12], MAGNET [13], require massive amounts of training data for model optimization, which inevitably leads to performance degradation in the few-shot scenario. However, the PLMs tuning multi-label text classification methods can provide a certain advantage in the few-shot problem. Therefore, three PLMs tuning methods are conducted as compared methods, the details are described as follows:

TextCNN [45]: The method uses a simple CNN with one layer of convolution on top of word vectors for Sentence Classification. In our experiments, PLMs are used to learn the representation of words, in addition, a multi-label classification layer is added to predict labels. Notably, the method is fine-tuned on the support set to select the optimal model and validated on the query set.

TagBert [46]: This is a model based on a large pre-trained model and a multi-label classification layer. Following the parameter setting of a threshold-based multi-label method, a fixed threshold tuned on the support set is used in the experiments.

BertFGM (https://github.com/percent4/keras_bert_multi_label_cls (accessed on 2 April 2021)): Based on the TagBert method, adversarial training [47] is introduced to increase the robustness and generalization of the model.

The experimental setup of all the above methods is the same as that in TextCNN.

4.2.2. Evaluation

In our paper, the *MacroF1* and *MicroF1* are introduced to evaluate the effectiveness of our proposed method. *MacroF1* calculates the average of the *F1* scores obtained for each category, which can be formulated as (5):

$$\begin{aligned} P_t &= \frac{TP_t}{TP_t + FP_t} \\ R_t &= \frac{TP_t}{TP_t + FN_t} \\ \text{Macro F1} &= \frac{1}{|C|} \sum_{t \in C} \frac{2P_t R_t}{P_t + R_t} \end{aligned} \quad (5)$$

where P_t represents the precision of each category, R_t represents the recall of each category. TP_t , FP_t and FN_t are the true-positive, false-positive and false-negative example of the t -th label in the label set C , respectively. *MicroF1* calculates the overall of the *F1* scores, which can be formulated as (6):

$$\begin{aligned} P &= \frac{\sum_{t \in C} TP_t}{\sum_{t \in Y} TP_t + FP_t} \\ R &= \frac{\sum_{t \in C} TP_t}{\sum_{t \in C} TP_t + FN_t} \\ \text{Micro F1} &= \frac{2PR}{P+R} \end{aligned} \quad (6)$$

where P represents the overall precision, R represents the overall recall.

4.3. Experimental Results

4.3.1. Experiment Settings

We evaluate the performance of our proposed method on the few-shot Exercises–Concepts dataset. Because some concepts in the dataset have only 5 exercises, we select the value K in K -shot as 1 and 5, respectively. There are some hyper-parameters that need to be initialized in the above methods. Firstly, we introduce uniform settings in all methods. The maximum length sequence is set as 512. These models are optimized by Adam with batch size 4 and learning rate 1×10^{-5} . Then, the size of thresholds has an impact on final performance. The thresholds are set as 0.10, 0.65, 0.82, 0.24 on 1-shot setting in TextCNN,

TagBert, BertFGM and PTMLTC, respectively. On 5-shot setting, the thresholds are 0.08, 0.70, 0.85 and 0.20. The reported results are the mean and variance of the experimental results on 10 randomly generated few-shot datasets.

4.3.2. Performance Comparison

Results of 1-shot setting: The results of the 1-shot exercise linking to knowledge concepts are shown in Table 2. From the experimental results, we can have the following observations. Firstly, we can observe that the results of *MicroF1* and *MacroF1* in the PTMLTC method are 54.74% and 46.11%, respectively, which are far better than the other three baselines. In the case of much training data, the performance of BertFGM is better than the TagBert. However, added adversarial training in the few-shot problem obtains interference information, which makes the classifier more indistinguishable. BertFGM achieves worse results than the TagBert. Results of 5-shot setting: The results of the 5-shot exercise linking to knowledge concepts are shown in Table 3. The results are basically consistent with the trend of the 1-shot setting. Compared with the 1-shot setting, the results of all methods have been improved in the 5-shot setting. These results demonstrated that the increasing of training data improves classification performance. In addition, PTMLTC has a smaller margin of advantage in 5-shot setting compared with 1-shot setting. It is proved that the fewer the data, the more obvious the advantages of PTMLTC.

Table 2. Results of 1-shot on our dataset. Metrics marked in bold contain the highest metrics for the dataset.

Method	1-Shot	
	Micro F1	Macro F1
TextCNN	6.60 ± 1.23	5.70 ± 0.89
TagBert	9.83 ± 0.77	6.05 ± 2.16
BertFGM	6.65 ± 2.47	2.11 ± 2.13
PTMLTC	53.86 ± 3.16	49.04 ± 3.42

Table 3. Results of 5-shot on our dataset. Metrics marked in bold contain the highest metrics for the dataset.

Method	5-Shot	
	Micro_F1	Macro_F1
TextCNN	29.49 ± 0.62	29.84 ± 2.67
TagBert	47.06 ± 0.18	41.50 ± 6.90
BertFGM	34.72 ± 0.99	26.66 ± 3.12
PTMLTC	62.37 ± 0.43	58.84 ± 0.84

4.3.3. Ablation Study

We compare the effects with different PLMs. In our proposed methods, Bert [15] and Roberta [48] models are adopted with bert-base-chinese (<https://huggingface.co/bert-base-chinese> (accessed on 5 February 2022)) and chinese-roberta-wwm-ext (<https://huggingface.co/hfl/chinese-roberta-wwm-ext> (accessed on 6 February 2022)). Table 4 summarizes The results are summarized in Table 4, which shows the Roberta-based pre-training model achieves better results than Bert.

The success of prompt tuning mainly owes to the template design and label words. Different templates are designed in our method to discuss their effect. The details are shown Table 5. The template was selected as “It belongs to [MASK]”, which obtains the better result.

Table 4. Results of different PLMs on our dataset. Metrics marked in bold contain the highest metrics for the dataset.

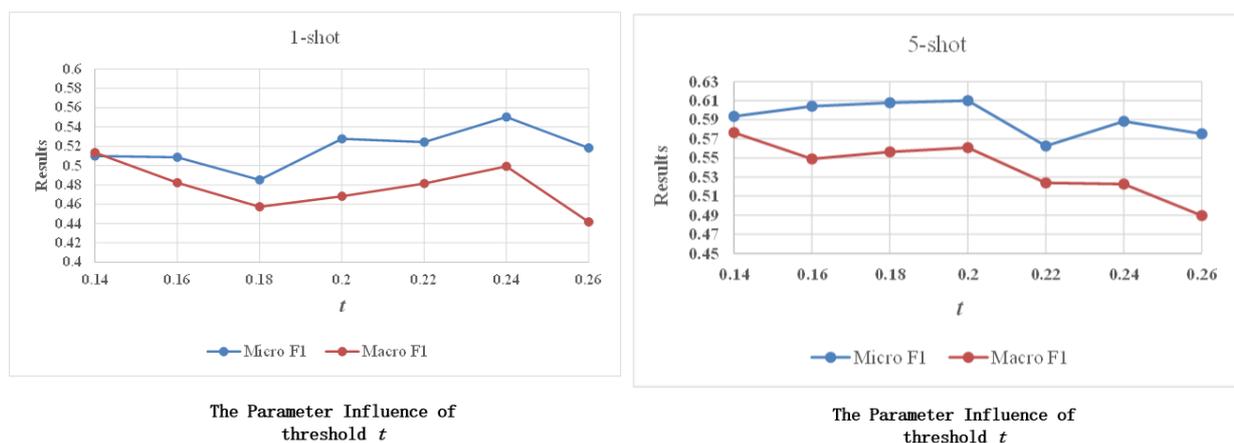
Method	Micro F1		Macro F1	
	1-Shot	5-Shot	1-Shot	5-Shot
PTMLTC_Bert	50.74 ± 1.53	58.56 ± 1.50	46.11 ± 2.68	54.28 ± 1.47
PTMLTC_Roberta	53.86 ± 3.16	62.37 ± 0.43	49.04 ± 3.42	58.84 ± 0.84

Table 5. Results of the different design of the templates. Metrics marked in bold contain the highest metrics for the dataset.

Templates	1-Shot		5-Shot	
	Micro F1	Micro F1	Micro F1	Micro F1
It belongs to [MASK].	53.86 ± 2.74	49.04 ± 3.12	62.37 ± 1.05	58.84 ± 0.89
The concept is [MASK].	50.98 ± 2.92	51.35 ± 2.15	58.44 ± 0.77	54.28 ± 1.15
The concept belongs to [MASK].	52.76 ± 3.13	46.83 ± 2.47	60.99 ± 0.37	53.85 ± 0.62

4.3.4. Parameter Sensitivity

Regarding our proposed method, in this section we have studied the influence of the parameter, which is the threshold t in Equation (3). The experimental mode of control variables is adopted, when one variable is changed, the other variables remain unchanged. We randomly selected a dataset from the 1-shot and 5-shot few-shot datasets for verification. After some preliminary tests, we found that the value of t will have a relatively large impact on the effect, it can be ensured that the effect will not excessively fluctuate within a certain range. The value set of t is [0.14, 0.16, 0.18, 0.22, 0.24, 0.26]. It can be observed from Figure 4 that $t = 0.24$ on the 1-shot setting and $t = 0.2$ on the 5-shot setting lead to the best results.

**Figure 4.** Effects of threshold t on two datasets.

5. Conclusions and Future Work

In this paper, a prompt tuning multi-label text classification method is proposed to realize the link between exercises and knowledge concepts. The main idea is that the relevance scores of exercise content and knowledge concepts are learned by a prompt tuning model with a unified template, and then the multiple associated knowledge concepts are selected with a threshold. On the constructed dataset, we compare the proposed method with other baseline methods. The results show that PTMLTC achieves better performance than other state-of-the-art methods in the evaluation metrics, and with fewer training data, the advantage is more conspicuous. The knowledge concepts in the course bear a natural

graph relationship, and our work ignores the relationship between them. Future work will try to introduce the structural relationship between knowledge concepts into the model for achieving better results.

Author Contributions: Methodology, Y.L.; Resources, B.L.; Validation, L.Z.; Writing—original draft, L.W.; Writing—review & editing, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

- Gong, J.; Wang, S.; Wang, J.; Feng, W.; Peng, H.; Tang, J.; Yu, P.S. Attentional graph convolutional networks for knowledge concept recommendation in moocs in a heterogeneous view. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, 25–30 July 2020; pp. 79–88.
- Chen, S.Y.; Wang, J.H. Individual differences and personalized learning: A review and appraisal. *Univers. Access Inf. Soc.* **2021**, *20*, 833–849. [\[CrossRef\]](#)
- Khalid, A.; Lundqvist, K.; Yates, A. Recommender systems for moocs: A systematic literature survey (January 1, 2012–July 12, 2019). *Int. Rev. Res. Open Distrib. Learn.* **2020**, *21*, 255–291.
- Corbett, A.T.; Anderson, J.R. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Model. User-Adapt. Interact.* **1994**, *4*, 253–278. [\[CrossRef\]](#)
- He, Z.; Li, W.; Yan, Y. Modeling knowledge proficiency using multi-hierarchical capsule graph neural network. *Appl. Intell.* **2021**, *52*, 7230–7247. [\[CrossRef\]](#)
- Okpo, J.; Masthoff, J.; Dennis, M.; Beacham, N. Conceptualizing a framework for adaptive exercise selection with personality as a major learner characteristic. In Proceedings of the Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, 9–12 July 2017; pp. 293–298.
- Okpo, J.A.; Masthoff, J.; Dennis, M. Qualitative Evaluation of an Adaptive Exercise Selection Algorithm. In Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, Utrecht, The Netherlands, 21–25 June 2021; pp. 167–174.
- Gao, W.; Liu, Q.; Huang, Z.; Yin, Y.; Bi, H.; Wang, M.C.; Ma, J.; Wang, S.; Su, Y. Rcd: Relation map driven cognitive diagnosis for intelligent education systems. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, 11–15 July 2021; pp. 501–510.
- Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L.J.; Sohl-Dickstein, J. Deep knowledge tracing. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montreal, QC, USA, 7–12 December 2015; Volume 28.
- Zhang, J.; Shi, X.; King, I.; Yeung, D.Y. Dynamic key-value memory networks for knowledge tracing. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 765–774.
- Ghosh, A.; Heffernan, N.; Lan, A.S. Context-aware attentive knowledge tracing. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, 6–10 July 2020; pp. 2330–2339.
- Liu, J.; Chang, W.C.; Wu, Y.; Yang, Y. Deep learning for extreme multi-label text classification. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 115–124.
- Pal, A.; Selvakumar, M.; Sankarasubbu, M. Multi-label text classification using attention-based graph neural network. *arXiv* **2020**, arXiv:2003.11644.
- Chang, W.C.; Yu, H.F.; Zhong, K.; Yang, Y.; Dhillon, I. X-bert: Extreme multi-label text classification with bert. *arXiv* **2019**, arXiv:1905.02331.
- Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
- Babbar, R.; Schölkopf, B. Dismec: Distributed sparse machines for extreme multi-label classification. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, Cambridge, UK, 6–10 February 2017; pp. 721–729.
- Yen, I.E.; Huang, X.; Dai, W.; Ravikumar, P.; Dhillon, I.; Xing, E. Ppdspare: A parallel primal-dual sparse method for extreme classification. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 545–553.

18. Prabhu, Y.; Kag, A.; Harsola, S.; Agrawal, R.; Varma, M. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 993–1002.
19. Jain, H.; Prabhu, Y.; Varma, M. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 935–944.
20. Akbarnejad, A.H.; Baghshah, M.S. An efficient semi-supervised multi-label classifier capable of handling missing labels. *IEEE Trans. Knowl. Data Eng.* **2018**, *31*, 229–242. [[CrossRef](#)]
21. Tagami, Y. Annxml: Approximate nearest neighbor search for extreme multi-label classification. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 455–464.
22. Prabhu, Y.; Varma, M. Fastxml: A fast, accurate and stable tree-classifier for extreme multi-label learning. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 263–272.
23. Qiang, J.; Chen, P.; Ding, W.; Wang, T.; Xie, F.; Wu, X. Heterogeneous-length text topic modeling for reader-aware multi-document summarization. *ACM Trans. Knowl. Discov. Data (TKDD)* **2019**, *13*, 1–21. [[CrossRef](#)]
24. Qiang, J.; Wu, X. Unsupervised statistical text simplification. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 1802–1806. [[CrossRef](#)]
25. Xiao, Y.; Li, Y.; Yuan, J.; Guo, S.; Xiao, Y.; Li, Z. History-based attention in Seq2Seq model for multi-label text classification. *Knowl.-Based Syst.* **2021**, *224*, 107094. [[CrossRef](#)]
26. Ma, Y.; Liu, X.; Zhao, L.; Liang, Y.; Zhang, P.; Jin, B. Hybrid embedding-based text representation for hierarchical multi-label text classification. *Expert Syst. Appl.* **2022**, *187*, 115905. [[CrossRef](#)]
27. Chen, G.; Ye, D.; Xing, Z.; Chen, J.; Cambria, E. Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In Proceedings of the 2017 IEEE International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 2377–2383.
28. Yang, P.; Sun, X.; Li, W.; Ma, S.; Wu, W.; Wang, H. SGM: Sequence generation model for multi-label classification. *arXiv* **2018**, arXiv:1806.04822.
29. Xun, G.; Jha, K.; Sun, J.; Zhang, A. Correlation networks for extreme multi-label text classification. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, 6–10 July 2020; pp. 1074–1082.
30. Ding, N.; Hu, S.; Zhao, W.; Chen, Y.; Liu, Z.; Zheng, H.T.; Sun, M. Openprompt: An open-source framework for prompt-learning. *arXiv* **2021**, arXiv:2111.01998.
31. Schick, T.; Schütze, H. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv* **2020**, arXiv:2001.07676.
32. Zhang, N.; Li, L.; Chen, X.; Deng, S.; Bi, Z.; Tan, C.; Huang, F.; Chen, H. Differentiable prompt makes pre-trained language models better few-shot learners. *arXiv* **2021**, arXiv:2108.13161.
33. Floridi, L.; Chiriatti, M. GPT-3: Its nature, scope, limits, and consequences. *Minds Mach.* **2020**, *30*, 681–694. [[CrossRef](#)]
34. Hu, S.; Ding, N.; Wang, H.; Liu, Z.; Li, J.; Sun, M. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. *arXiv* **2021**, arXiv:2108.02035.
35. Chen, X.; Zhang, N.; Xie, X.; Deng, S.; Yao, Y.; Tan, C.; Huang, F.; Si, L.; Chen, H. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. *arXiv* **2021**, arXiv:2104.07650.
36. Ma, Y.; Wang, Z.; Cao, Y.; Li, M.; Chen, M.; Wang, K.; Shao, J. Prompt for Extraction? PAIE: Prompting Argument Interaction for Event Argument Extraction. *arXiv* **2022**, arXiv:2202.12109.
37. Cui, L.; Wu, Y.; Liu, J.; Yang, S.; Zhang, Y. Template-based named entity recognition using BART. *arXiv* **2021**, arXiv:2106.01760.
38. Li, X.L.; Liang, P. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv* **2021**, arXiv:2101.00190.
39. Scao, T.L.; Rush, A.M. How many data points is a prompt worth? *arXiv* **2021**, arXiv:2103.08493.
40. Hambarzumyan, K.; Khachatryan, H.; May, J. Warp: Word-level adversarial reprogramming. *arXiv* **2021**, arXiv:2101.00121.
41. Reynolds, L.; McDonnell, 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.
42. Xu, G.; Lee, H.; Koo, M.W.; Seo, J. Convolutional neural network using a threshold predictor for multi-label speech act classification. In Proceedings of the 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju, Korea, 13–16 February 2017; pp. 126–130.
43. Hou, Y.; Lai, Y.; Wu, Y.; Che, W.; Liu, T. Few-shot learning for multi-label intent detection. *arXiv* **2020**, arXiv:2010.05256.
44. Yu, J.; Luo, G.; Xiao, T.; Zhong, Q.; Wang, Y.; Feng, W.; Luo, J.; Wang, C.; Hou, L.; Li, J.; et al. MOOCube: A large-scale data repository for NLP applications in MOOCs. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5 July 2020; pp. 3135–3142.
45. Chen, Y. Convolutional Neural Network for Sentence Classification. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
46. Khezrian, N.; Habibi, J.; Annamoradnejad, I. Tag recommendation for online Q&A communities based on BERT pre-training technique. *arXiv* **2020**, arXiv:2010.04971.

-
47. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
 48. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.