Appendix: An Empirical Study of Finding Similar Exercises

Anonymous Author(s)
Affiliation
Address
email

A Appendix

A.1 Data Sets.

The anonymous education platform[1] contains millions of exercises and we only choose about 350K junior math exercises for our experiments. We sample 1500 as seed exercises and construct 23K exercise pairs through the BM25 match and some strategies such as random choose and random with concept. Then theses exercises are labeled with several similar exercises and each given exercise is labeled by three teachers. We choose the majority numbers of votes as the label for the similar exercise. We split our data set randomly via the seed exercises into three parts: 80% is for training set, 10% is for validation set and 10% is for test set. Finally, we only report the performances on the test set.

A.2 Experimental Settup

We implement all the models with Tensorflow in our experiments. In the pre-training stage, we pre-train our models with MLM objective, continuing from the published checkpoint, BERT-base-chinese. We pre-train our model for 200K steps, and the first 3000 steps are for warm-up. The rest of the hyper-parameters are the same as BERT-base. In the fine-tuning stage, we train our model in the multi-task paradigm. The multi task module adopts three layers of neural network, and the output sizes of hidden layer of each layer are 768,768 and 3. We apply the Adam method to optimize our model. The learning rate is $2e-5$, the number of training epoch is 3. We conduct our experiments with 2 Tesla T4 GPUs.

B Implementation of MoE Layer

We adopt a 3-layer neural network to dynamically learn the coefficients of tasks which is similar to MoE Layer[**?**] in the information recommendation field. The detail operations are as follows: First of all, we concat the feature representations of the different tasks:

$$feature = concat(Fe_{T_1}, Fe_{T_2}, Fe_{T_3}) \tag{1}$$

Secondly, for the feature representation, we learn the parameter coefficients through a three-layer neural network.

$$\alpha = (\alpha_1, \alpha_2, \alpha_3) = concat(Gate_1(Fe_{T_1}), Gate_2(Fe_{T_2}), Gate_3(Fe_{T_3})) \tag{2}$$

where $Gate_i(.)$ is the i-th expert network with a three-layer neural network.

Finally, we assign different task weights to different tasks as the above Formula-**??**.

[1] Anonymous education platform: https://anonymous.com

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Workshop on Math AI for Education (MATHAI4ED). Do not distribute.

C Problem Formulation

As mentioned earlier, similar exercises are those having the same purpose which is related with the semantics of exercises.

Definition 1. *Given a set of exercises including stem, option, concept of knowledge and exercise analysis, our target is to learn a model $\mathcal{F}$ which can be used to measure the similarity scores pairs and find similar exercises for any exercise E by ranking the candidate ones $\mathcal{D}$ with similarity scores:*

$$\mathcal{F}(E, \mathcal{D}, \Theta) \rightarrow \mathcal{R}^s \tag{3}$$

where Θ is the parameters of $\mathcal{F}$, $\mathcal{D} = (E_1, E_2, E_3, \cdots)$ are the candidate exercises for E and $\mathcal{R}^s = (E_1^s, E_2^s, E_3^s, \cdots)$ are the candidates ranked in descending order with their similarity scores $(S(E, E_1^s), S(E, E_2^s), S(E, E_3^s), \cdots)$. The similar exercises for E are those candidates having the largest similarity score.

D Testing

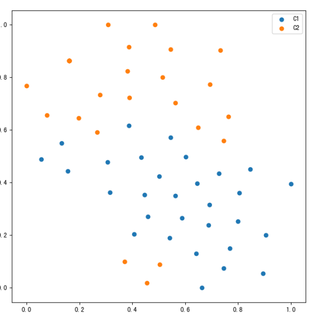
After obtaining the trained ExerciseBERT, for any *exercise E* in the testing stage, we could find its similar exercises by ranking the candidate ones according to their similarity scores, and finally return the accurate Top-K similar exercises. We use the Precision@K as the metric. Precision@K is calculated as follows:

$$Precision@k = \sum_{i=1}^{N} \frac{true\ positives\ @k}{(true\ positives\ @k) + (false\ positives\ @k)} \tag{4}$$

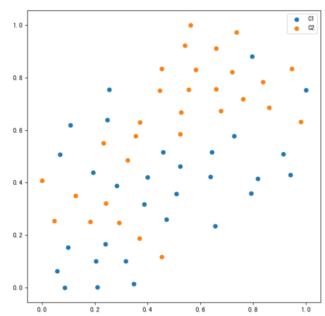
where k=1, 3 and 5 and N is the number of seed exercises.

E Visualization Analysis

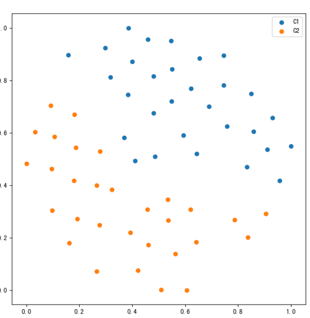
We conduct visualization analysis of the ExerciseBERT's representation. It is important to learn exercise representations in which similar exercises are closer while dissimilar exercises are farther. To show the results intuitively, we first select four groups of exercises under two concepts that are randomly selected(the first two groups have the close knowledge concepts while the latter two are different), and then reduce the dimension of obtained representations by t-SNE. The visualization results are shown in Figure-1 and we can get two interesting phenomena. On the one hand, if the knowledge concepts of C1 and C2 are relatively similar (Fig-1(a) and Fig-1(b)), their exercises representation are also relatively close. There may be some overlapping parts because similar exercises are those having the same purpose including not only knowledge concept but also some other information such as problem-solving ideas. On the other hand, if the knowledge concepts of C1 and C2 are quite different (Fig-1(c) and Fig-1(d)), they are unlikely to become similar exercises and their exercises representation have a big difference. Thus, the results show that ExerciseBERT has a good exercise representation.



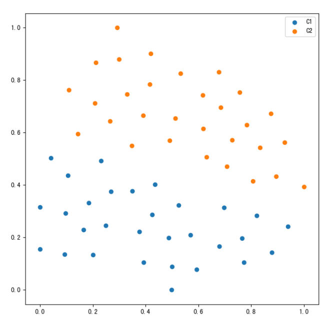
(a) Group 1



(b) Group 2



(c) Group 3



(d) Group 4

Figure 1: Visualization Of ExerciseBERT Representation