Sequence Generation Model Integrating Domain Ontology for Mathematical question tagging

TAO HUANG, Central China Normal University, China
SHENGZE HU*, Central China Normal University, China
KEKE LIN, Central China Normal University, China
HUALI YANG, Wuhan Textile University, China
HAO ZHANG*, Central China Normal University, China
HOUBING SONG, Embry-Riddle Aeronautical University, USA
ZHIHAN LV, Uppsala University, Sweden

In online learning systems, tagging knowledge points for questions is a fundamental task. Automatic tagging technology uses intelligent algorithms to automatically tag knowledge points for questions to reduce manpower and time costs. However, the current knowledge point tagging technology cannot satisfy the situation that mathematics questions often involve a variable number of knowledge points, lacks the consideration of the characteristics of the mathematics field, and ignores the internal connection between knowledge points. To address the above issues, we propose a Sequence Generation Model Integrating Domain Ontology for Mathematical question tagging (SOMPT). SOMPT performs data augmentation for text and then obtains intermediate text based on domain ontology replacement to facilitate deep learning model to understand mathematical question text. SOMPT is able to obtain dynamic word vector embedding to optimize the textual representation for math questions. What's more, our model can capture the relationship between tags to generate knowledge points more accurately in the way of sequence generation. The comparative experimental results show that our proposed model has an excellent tagging ability for mathematical questions. Moreover, the sequence generation module in SOMPT can be applied on other multi-label classification tasks and be on par with the state-of-the-art performance models.

CCS Concepts: • **Computing methodologies** → **Language resources**; **Natural language processing**; **Natural language generation**; • **Applied computing** → **Computer-assisted instruction**.

Additional Key Words and Phrases: Mathematical question tagging, Deep learning, Language models, Sequence generation

1 INTRODUCTION

With the construction of various intelligent education platforms and the continuous acquisition of various learning data, personalized adaptive learning is increasingly becoming the focus of the education field [21, 26]. Education resources are the core of the personalized adaptive learning framework. They are widely used in cognitive

*Corresponding author

Authors' addresses: Tao Huang, tmht@mail.ccnu.edu.cn, Central China Normal University, Wuhan, China, 430079; Shengze Hu, hsz666@mails.ccnu.edu.cn, Central China Normal University, Wuhan, China, 430079; Keke Lin, linkeke@mails.ccnu.edu.cn, Central China Normal University, Wuhan, China, 430079; Huali Yang, yanghuali@mail.ccnu.edu.cn, Wuhan Textile University, Wuhan, China, 430200; Hao Zhang, zhanghao@mail.ccnu.edu.cn, Central China Normal University, Wuhan, China, 430079; Houbing Song, Houbing.Song@erau.edu, Embry-Riddle Aeronautical University, Daytona Beach, USA, 32114-3900; Zhihan Lv, lvzhihan@gmail.com, Uppsala University, Uppsala, Sweden, 75104.

2375-4699/2023/4-ART $15.00
https://doi.org/10.1145/3593804

diagnosis [54], knowledge tracing [18, 32] and personalized educational learning resource recommendation [28, 31], which all require resources to be organized and related in some way. Among all education resources [13, 15, 34], questions dominate and are in great demand in students' daily learning, and it becomes necessary to create a good index structure for questions by tagging knowledge points.

The subject questions are mainly manually tagged by hiring experts, but this approach requires a lot of investment in manpower and there are limitations in the consistency, credibility, and later maintenance and update of knowledge points. Fortunately, some progress has been made in automatic tagging. The automatic knowledge point tagging can automatically identify the knowledge points investigated by the questions through the intelligent algorithm, and label the relevant knowledge points for them. Since the tagging process is often completed by the same model, the standards are consistent and the consistency of tagging results is thus guaranteed. In addition, a large number of questions are tagged by high-power machines, which can greatly reduce manpower and time costs. This paper is devoted to the research of knowledge point tagging automation technology, which is applied in the field of mathematics.

However, there are still some challenges in the automatic knowledge point tagging for mathematics questions. On the one hand, most existing automatic tagging methods regard the knowledge point tagging task as a multi-classification task [3, 23]. In a single-label multi-classification task, a sample can only belong to one label, which cannot meet the requirements of mathematics test labeling. Mathematical questions often involve a variable number of knowledge points, as shown in Figure 1. Therefore, we regard the knowledge point tagging task as a multi-label multi-classification task [17, 35]. In a multi-label classification (MLC) task, a sample can be tagged as multiple labels with an unfixed number, and there is no mutually exclusive relationship between labels. However, there are still relatively few studies on applying MLC technologies to knowledge point tagging.

On the other hand, the general classification methods focus mainly on the implementation and improvement of algorithms and lack the consideration of the uniqueness of mathematical domain. As a vehicle for practicing and testing knowledge in the mathematical domain, the mathematical questions are described in a variety of ways. A large number of mathematical symbols and formula descriptions will produce a large number of entity expressions, which will lead to problems such as sparse data and ambiguity when extracting text feature vectors of mathematical problems in deep learning models, and then lose the inherent semantic related information of the question. Some researchers have tried in this regard and proposed automatic labeling techniques for mathematical problems to deal with and study the particularity of textual descriptions of mathematical problems, such as a set of novel probabilistic latent class models [5] and classification methods based on text combined with mathematical expression structures [38]. However, it is difficult for current models to consider the connections between knowledge points while capturing semantic information about the questions. Mathematics questions often involve multiple knowledge points, and these knowledge points often have related relationships, such as predecessors and successors. Therefore, we need to consider the characteristics of the field of mathematics. We not only need to deal with the texts, symbols, and formulas in the questions in a specific way, but also try to capture the internal connections between the knowledge points.

To address the above issues, this paper proposes to pre-process the text and obtain an intermediate text based on ontology replacement, considering the specificity of the mathematical domain comprehensively which include richer textual information. In order to ensure uniform tagging standards during the tagging process, we regard the task of automatic knowledge point tagging of math questions as a multi-label classification task for the text content of math questions. Taking the significance of the text of mathematical questions for knowledge points prediction and the necessity of linkage between knowledge points into account, this paper uses a sequence generation approach based on UniLM (Unified Language Model Pre-training for Natural Language Understanding and Generation) [11] to predict knowledge points for questions.UniLM is based on BERT (Bidirectional Encoder Representation from Transformers) [19] model which make it can obtain a dynamic word vector representation after pre-training. And the UniLM model uses three special Mask as pre-training objectives, so that the model can

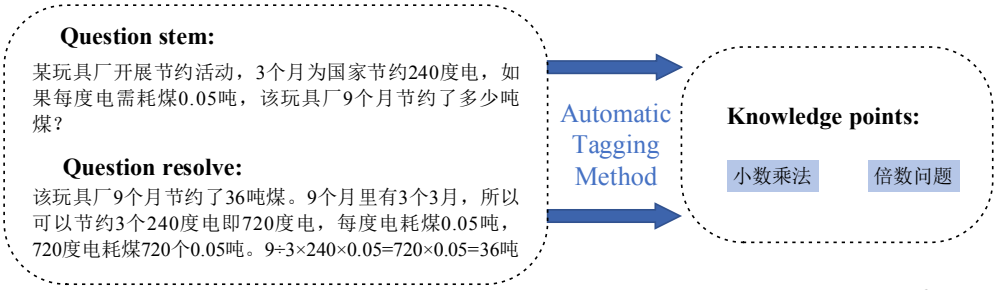

Fig. 1. An example data for automatic tagging knowledge points for mathematics questions.

be applied to NLG tasks and achieve the same effect as BERT in NLU tasks. We choose one of them, the sequence-to-sequence (Seq-to-Seq) mask, as the mask mode of self-attention mechanism [41], and construct a sequence generation model based on UniLM, which can fully extract text information of the mathematical questions as well as draw the association relationship of knowledge points. All in all, we propose a sequence generation model integrating domain ontology for mathematical question tagging (SOMPT). The main contributions of this paper are summaries as follows.

(1) We propose a data augmentation method based on text replacement, which is to obtain the intermediate text of questions based on integrating domain ontology and replacing named entity alleviating data sparsity in view of the importance of mathematical ontology and the diversity of representation forms.

(2) A novel sequence generation method is proposed to tag mathematical questions with knowledge points. It is based on the UniLM to obtain richer context representation and uses the Seq-to-Seq mask attention mechanism to generate knowledge points which can capture the relationship between knowledge points.

(3)SOMPT not only performs well on the standard dataset for the study of mathematical question tagging. The Knowledge points prediction module based on sequence generation also can be applied to the task of multi-label text classification, and the classification performances on some publicly available datasets are also quite good.

The remainder of the paper is organized as follows. Section 2 reviews relevant work in recent years. Section 3 details the proposed mathematical question tagging model. Experimental results and analyses are presented in Section 4. Finally, we summary our work in Section 5.

2 RELATED WORKS

2.1 Automatic tagging for educational resources

The variety and quantity of educational resources in each subject area is large and is organized mainly through knowledge points from different disciplines. Knowledge points are often used as an important research basis for learning resource recommendation. In studying personalized question recommendation methods, literature [42] considers the weight of knowledge points, and literature [46] explores the associations between knowledge points. There have also been applications of knowledge graphs to mining and recommending educational resources by studying knowledge points. Literature [17] applies neural networks to extract pedagogical concepts from instructional data and automatically construct an educational knowledge graph. Literature [7] proposes a personalized recommendation model for diverse online resources by analyzing students' mastery of knowledge points and found that constructing a knowledge graph organizes knowledge points well. The literature [22] proposes a model for recommending personalized knowledge points by mapping knowledge points to a knowledge graph.

The tagging of educational resources is indispensable in personalized adaptive learning. Nowadays, resources in the field of education are becoming more and more diversified, and the forms shown by educational resource tagging have become more diverse. Literature [36] uses a document extraction attention network for thematic tagging to analyze students' abilities and recommend relevant reading materials. Literature [1] studies a method that automatically recommends tags for students' questions in a community Q&A system to coordinate communication between teachers and students.

As an important part of educational resources, questions also need to be tagged with knowledge points through tagging technology. The knowledge point tagging for the question is mainly tagging for the text of the question. For text tagging, some researchers have focused on text ontologies, literature [9] proposes document semantic tagging improvement methods using the semantic environment information expressed by domain ontologies. The importance of ontology is also mentioned by literature [50] in a review of semantic tagging of text; literature [8] proposes text tagging algorithm is based on domain ontology; literature [49] constructs a semantic ontology knowledge base to infer the relationship between entities and then tagged semantic according to semantic relations. To enhance knowledge point tagging of questions, deep learning techniques are widely used: literature [17] designs an expertise-enriched Convolutional Neural Networks(CNN) model; Literature [35] presents a location-based attention model and a keyword-based model to automatically tag questions; Literature [10] introduces a method based on ensemble learning. To support teachers' teaching and students' learning more effectively, it is necessary to further improve the automatic tagging technology of educational resources, so that they can be well organized for personalized adaptive learning.

2.2 Multi-label text classification

This paper mainly adopts natural language processing (NLP) technology for automatic question tagging task of mathematical questions as a text multi-label classification task. Considering mathematical questions are often associated with multiple knowledge points, the knowledge points tend to be relatively fixed and uniform. Therefore, it is most common to treat the automatic tagging task as a multi-label classification task. The simplest idea is the "one-vs-all" scheme for MLC, which is transformed into multiple single-label binary classification tasks, using a binary classifier to predict 0 or 1 for each category [3, 23]. This thought is based on assumption that there is no association between tags, yet in reality multiple tags are often connected and complex. According to the disadvantages mentioned above, some machine learning researches have made improvements to use the association between labels as a classification reference. The specific approach is to predict the current label considering not only textual features, but also the previous predicted label. Literature [47] proposes a novel MLC model that combines rank support vector machines and binary correlation with robust low-rank learning to overcome, as much as possible, the two drawbacks of inter-class imbalance and ignoring label correlation that exist in binary correlation.

Deep learning is more comprehensive in extracting semantic information and can incorporate more label associations in predicting labels. Some researchers significantly improved multi-label text classification using CNN [2, 20, 43]. Compared with traditional machine learning, CNN can capture different levels and deeper semantic information through sliding windows. Literature [20] proposes parallel CNN and deep CNN to capture local semantic features respectively, and uses a max over time pooling layer to extract global semantic features. Literature [43] also applies CNN to extract local features. Literature [27] further introduces a self-attention based on CNN with better classification performance. In order to extract richer semantic information of the text, literature [6] proposes a model combining recurrent neural networks (RNN) and CNN for more fine-grained text MLC tasks. Literature [48] employs sequence generation ideas and applied an encoder-decoder model based on the long short-term memory (LSTM)[16]. Their experimental results show that this method not only captures associations between labels, but also automatically selects the most informative labels.

2.3 Sequence generation for multi-label classification

Natural language understanding (NLU) and natural language generation (NLG) are two core tasks of NLP. The goal of NLU is to enable computers to understand natural language (human language), etc., focusing on understanding [19, 29]. In other words, NLU takes natural language as input and outputs machine-readable semantic representation. NLU usually includes part-of-speech tagging, representation learning, text classification, etc. The goal of NLG is to express semantic information in human-readable natural language form. In other words, the input of NLG is data in a non-linguistic format, and the output is a language format that humans can understand. NLG mainly includes generation tasks such as machine translation and summary extraction. In this paper, we use the idea of sequence generation to accomplish the automatic tagging with knowledge points for mathematical questions. We transform the knowledge point multi-label classification task into the task of generating label text.

This approach first appeared in 2017 when literature [30] used a sequence-to-sequence model based on RNN to capture labels the correlation between labels and proposed to consider the sequence label prediction of a given text as MLC task. Subsequently, more new models or methods based on sequence generation to handle MLC tasks have been proposed. For example, literature [6] determines the "initial state" or prior knowledge of an RNN according to the feature vectors extracted by CNN and predicted label sequences. Literature [33] analyzes the implausibility of previous RNN models for predicting multiple labels in terms of training and prediction objectives and proposed new objectives based on the principle concept of ensemble probability. Literature [25] introduces a hybrid attention mechanism for generating higher-level semantic unit representations for MLC.

The sequence generation methods mentioned above are generally structured by RNN which will ignore the effect of the latter label on the labels already generated before it. To solve this challenge, literature [24] adds a fully connected layer to the part of predicting labels. Moreover, most models are limited in extracting text semantic information by CNN, so we propose using the UniLM model, which can extract more text semantic information and generate predicted labels with extracted label relevance. UniLM proposed by Microsoft [11], based on BERT [19]. By designing skillful masks for attention mechanism, multiple language models are integrated to satisfy both NLU and NLG. In this paper, the sequence generation idea is used as the main implementation of MLC and applied to knowledge point tagging, which is a novel technology discovery.

3 METHOD

The SOMPT framework is shown in Figure 2, which includes three parts: Ontology-based text replacement, Short text replacement for labels, and Knowledge points prediction based on UniLM for mathematical questions in Chinese.

3.1 Problem description

The knowledge points automatic tagging refers to automatically match to correct multiple knowledge points for questions based on the knowledge system. The model needs to try to understand the semantics of mathematical questions and extract knowledge information contained in questions to predict the knowledge points to which the questions belongs with maximum probability. It can be described formally as follows:

Definition: Assume that the dataset has m samples of math question and the sample space is $T = \{t_1, t_2, ..., t_m\}$. There are n knowledge points in total, and the set of knowledge points is $L = \{l_1, l_2, ..., l_n\}$. Each question t_i has multiple knowledge points and is represented as a set L'(as Figure 1). Given a mathematical questions dataset $D = \{(t_i, L'_i) \mid 0 \leq i \leq m, L'_i \in L\}$. This study uses D to construct an automatic model and achieve automatic tagging function for mathematical questions.

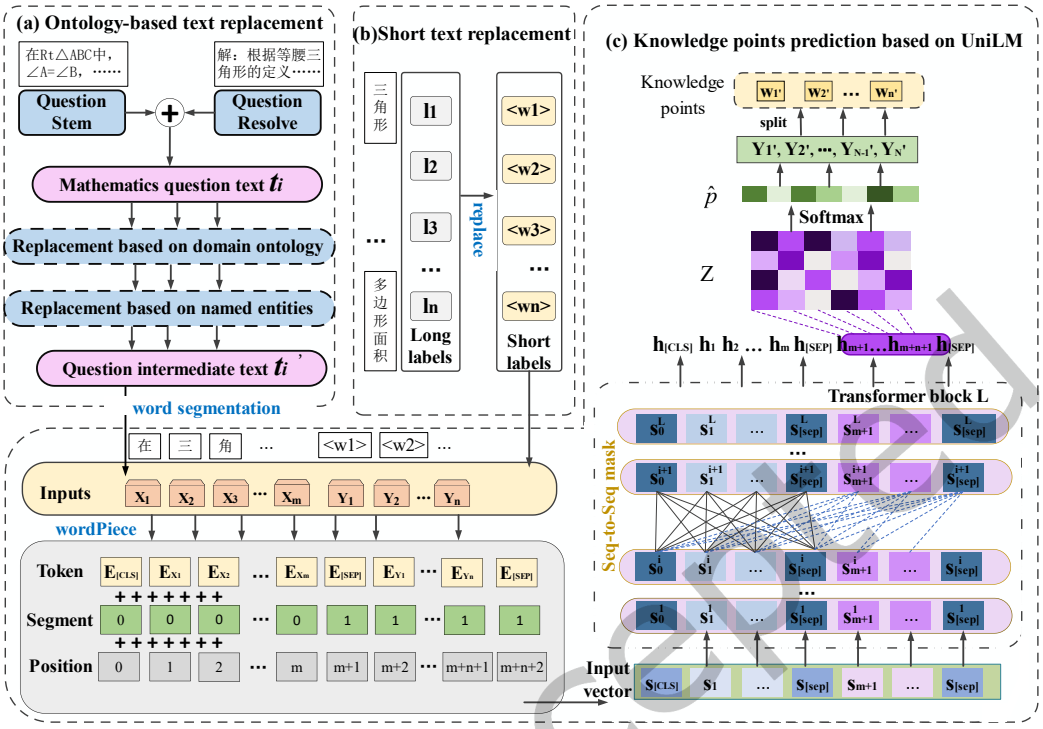

Fig. 2. The Framework of SOMPT. It consists of three parts: (a)Ontology-based text replacement, (b)Short text replacement, and (c)Knowledge points prediction based on UniLM.

3.2 Data augmentation based on ontology replacement

There are many entities in mathematics, and the same entity can have different concrete manifestations. For example, the entity of triangle may appear in the form of △, triangle, ABC, DEF, triangle ABC and so on in math test questions. For the knowledge point tagging task, the model needs to focus more on the concept of the triangle itself rather than the specific form of the triangle. Therefore, we need to do some pre-processing work on the text of math questions to avoid interference with the knowledge point annotation model. We stitch the text of the question stem and resolve together. We convert non-textual information to text or remove them, such as image links and formulas from the question. For some specific mathematical symbols, ontology replacement is performed with Chinese descriptions, to reduce data noise. The following text data augmentation strategy of replacement based on domain ontology and named entities is proposed to obtain the intermediate text of mathematical questions. Suppose that the text of the math question is t. The output of the converted intermediate text is t'. The process of acquiring intermediate text for mathematical questions can be represented as Transformation $(t_1, t_2, \ldots, t_m) \rightarrow (t'_1, t'_2, \ldots, t'_m)$.

3.2.1 ***Replacement based on domain ontology*** . Domain ontologies are described as domain entity concepts and their interrelationships, domain activities, properties and laws that the domain has, with a certain hierarchical structure.[52] In order to characterize numerous entities derived from the mathematical domain ontology in mathematical questions, we extract the intermediate text that integrating domain ontology. The specific process is as follows.

(1) The construction of a mathematical domain ontology. Mathematics domain ontology is constructed manually based on the content of teaching.[52] Firstly, we find out the core ontologies in mathematical concepts and list the important terms in the mathematical domain ontology. Then we define the classes, class hierarchies, the relevant properties of the classes and the relationships with other classes. The "class" represents a core concept in the field of mathematical domain, which has attributes and derives entities. Finally, we create instances and create an ontology library. As shown in Figure 3 for the partially constructed ontology, we can see that the domain ontologies have different levels. For example, the triangle is the uppermost layer and is at the core, while the acute triangle and equilateral triangle are at its lower level, and the equilateral triangle also belongs to acute triangle. Therefore, the variety of mathematical ontology will further aggravate the diversity of entities. In the knowledge point tagging task, the model should pay more attention to the ontology involved in the mathematics questions rather than specific entities. Therefore, based on the constructed mathematics domain ontology, it is necessary to deal with the related entities contained in the mathematics questions.

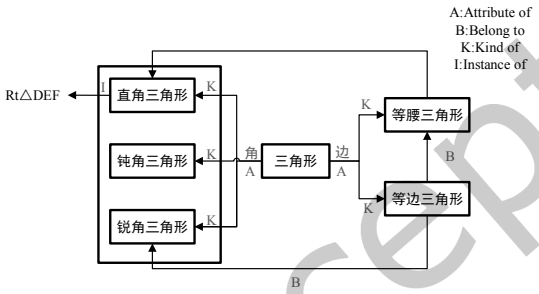

Fig. 3. Triangle ontology construction. Note:In this figure, “三角形” is at the core of domain ontology. “三角形” has the attributes of “角” and “边”, and can be derived into different entity concepts according to different division methods.

(2) Matching templates for entity recognition. The hierarchical expression of the domain ontology can enrich the semantic information of mathematical texts from multiple perspectives, so the entities in the texts need to be identified. We design matching templates for a limited number of elementary school mathematical question entities. We summarize the laws of hierarchical expressions based on the relationship between the elements in mathematical domain ontology library. Design regular expression matching templates according to different expressions of entities in the questions. According to the different expressions of entities in mathematical questions to design regular expression matching templates. We use different identification words to distinguish between the different classes of entities. Table 1 shows some samples of the recognition templates constructed in this paper.

(3) Uniform replacement based on domain ontology. The unified replacement based on domain ontology is divided into two processes: ontology replacement and affiliation tagging. The ontology replacement phase identifies and replaces entities and attributes in the question in the order from entity to attribute. Entities and attributes are replaced with new expression (The combination of the identification corresponding to the ontology in Table 1 and the sequential number that entity appears for the first time in the question.), and save their affiliation relationships information before replacement. The affiliation tagging phase iterates through the extracted expression of attributes and adds the dependencies based on the inclusion relationships between ontologies in the ontology library.The pseudocode of the replacement algorithm based on domain ontology is shown in Algorithm 1. For the text after this replacement process, we will use the Chinese description of ontology to replace the previous English replacement content. Figure 4 shows a sample.

Table 1. Entity identification templates.

ontology	regularexpression	identification
三角形	“三角形[A-Z]{3}”	triangle
四边形	“正方形[A-Z]{4}”; “长方形[A-Z]{4}”; “四边形[A-Z]{4}”; “菱形[A-Z]{4}”	quadrilateral
线	“直线[A-Z]{2}”; “射线[A-Z]{2}”; “线段[A-Z]{2}”	line
角	“角[A-Z]{3}”; “(?<![A-Z])[A-Z]{2}(?![a-zA-Z])”	angle
点	“点[A-Z]{1}”; “(?<![a-zA-Z])[A-Z]{1}(?![a-zA-Z])”	point

Algorithm 1 Construct the intermediate text for mathematical questions based on ontology

Input: Mathematical question,text;Template dictionary,ontologyDict
Output: Intermediate representation of mathematical question,text

function OntologyTransformation(text,OntologyDict)
  $ReplaceDict \leftarrow NULL$
  $OntologyTmp \leftarrow NULL$
  for OntologyKey,OntologyValue in OntologyDict **do**
    $index \leftarrow 1$
    Use OntologyValue to match text to get OntologyTmp
    Store OntologyTmp as the key, and the combination of OntologyKey and index as the value in ReplaceDict
    $index \leftarrow index + 1$
  end for
  for OntologyTmp, OntologyKey+index in ReplaceDict **do**
    if OntologyTmp is line/angle/point **then**
      Judge the ownership of triangles and quadrilaterals in OntologyTmp and ReplaceDict
      Add affiliation in text
    end if
  end for
  return text
end function

3.2.2 ***Replacement based on named entities.*** The mathematical questions exclude some conceptual entities in the specific domain of mathematics, and there are many entities in the general domain with different expressions. Therefore, based on the question text obtained from Section 3.2.1, we further process the text by identifying and replacing named entities, so that entities are renamed in the same way to make the text data neat. We use named entity recognition tool (Stanfordcorenlp[1]) to extract two major categories (entity category and time category)

[1]https://stanfordnlp.github.io/CoreNLP

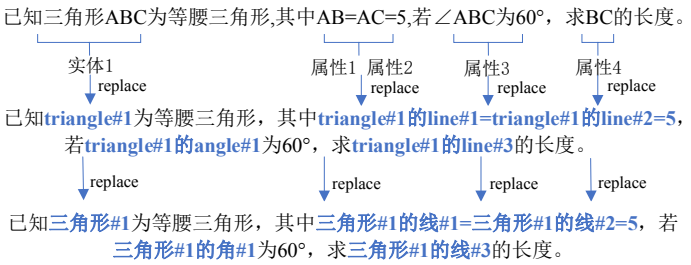

Fig. 4. A sample replacement question based on domain ontology. We judge the properties of entities according to their letter combinations.This question involves the ontology “三角形”, the entity concept “三角形ABC”, 等腰三角形”, the side length properties “AB” “AC” “BC”, and the angle property “∠ABC”.

and seven minor categories (person name, institution name, place name, time, date, currency, and percentage) of named entities from the text. Then all the recognized entities are named using a unified naming format, which is the entity category plus the sequential number in which the entity appears in text. An example of replacement based on named entities is shown in Figure 5.

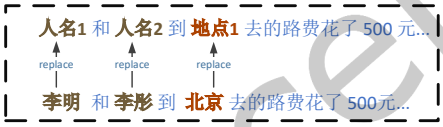

Fig. 5. A sample of Replacement based on named entities.

3.3 Replace knowledge points with short text

Our model mainly uses to the module used for sequence generation in the UniLM model to generate the knowledge points sequence. It is necessary to do some pre-processing work on the original knowledge points. Each knowledge point in Chinese math questions is generally composed of multiple Chinese characters, as shown in Figure 1. The text sequence composed of multiple knowledge points of each question will be a long text sequence. In the sequence generation model, the longer the generated target sequence, the lower the possibility of being completely correct and the lower the accuracy. Therefore, shortening the text sequence of the generated knowledge point labels can reduce the difficulty of correct prediction to a certain extent. In addition, each knowledge point label is a string composed of multiple characters in meaningful order. If the model needs to predict each character, then the prediction error of a single character will lead to the prediction error of the entire knowledge point string. Therefore, we designed the short text replacement to improve the tagging accuracy of the model. Specifically, we regard each knowledge point label as an independent and complete individual, and replace the entire character string of the knowledge point with one character. Considering that the characters used for replacement represent knowledge points rather than their original meanings, we choose characters that have not been used in all samples for replacement. We first count the token set of the question text after the word segmentation, and second, we select the tokens that do not appear in the set as the replacements of the label from the dictionary. In other words, the set of question characters and the set of label characters are completely non-overlapping, so as to avoid the ambiguity in label meaning. Finally, we replace the original labels with these selected characters. Part (b) in Figure 2 is this substitution process.

3.4 Multi-knowledge point tagging model

UniLM uses shared transformer network to achieve unified modeling as unsupervised pre-training [11]. In UniLM, different self-attention mask matrices are used for the language model (LM) objectives (i.e., bidirectional LM, unidirectional LM, and sequence-to-sequence LM). The bidirectional LM corresponds to an all-zero mask matrix, which allows word tokens to access the context. The left-to-right LM corresponds to an upper triangular masking matrix, i.e. allowing word tokens to see the preorder context. And the right-to-left LM corresponds to a lower triangular masking matrix. The sequence-to-sequence LM corresponds to a special mask matrix, as shown in Figure 6. There are the source sequence and the target sequence in the Seq-to-Seq LM. In the source sequence, word tokens can access the context, but not the target sequence. In the target sequence, word tokens follow the causality, that is, the following cannot be accessed. Thus, we use the source sequence to generate the target sequence. Considering that the number of knowledge points in the tagging task is uncertain, we can regard the knowledge point labels to be predicted as the target sequence. Specifically, we decide to transform the question text into the source sequence (S_1) and the knowledge points text into the target sequence(S_2), and use the sequence generation pattern in UniLM to predict sequences with knowledge points information. These sequences can be processed to obtain the knowledge points predicted by the model. In the process of sequence generation, the current generated content will participate in the next content generation. We believe that knowledge point tagging in this way can capture the connection between knowledge points, so as to improve the accuracy and credibility of knowledge point prediction [48]. Therefore, the principal body of SOMPT utilizes the Seq-to-Seq module in UniLM to complete the knowledge point tagging task, as shown in part (c) of Figure 2. The details of the model are described below.

Input embedding: We assume that every mathematical question text t_i' obtained by ontology-based replacement. We do Chinese word segmentation for t_i' to get a question text sequence $X = (x_1, x_2, ..., x_m)$ using the tokenizer function in the BERT4keras library[2]. The tokenizer function can divide the sentences into tokens using WordPiece algorithm, the same as in BERT [19]. The knowledge points text sequence is the new sequence $Y = (y_1, y_2, ..., y_n)$ that the knowledge points set $L_i' = (l_1, l_2, ..., l_n)$. n represents the number of knowledge points that the question t_i' has. The knowledge points text sequence is the new sequence after short text replacement. We splice the two text sequences with the [SEP] token to obtain the sequence

$$S = ([CLS], x_1, x_2, ..., x_m, [SEP], y_1, y_2, ..., y_n, [SEP]). \tag{1}$$

It is denoted as $S = (S_1, S_2)$, where [CLS] indicates the start of the sequence, and [SEP] indicates the end of the sequence. These two markers are also involved in model training, enabling the model to learn when to end the sequence generation process. S_1 is the source sequence and S_2 is the target sequence. We use S as the input sequence of the model. The text sequence S of each sample is further transformed to obtain word embedding by using the WordPiece algorithm, and then obtain their token embedding, position embedding and segment embedding. The input vectors s of the model is the sum of these three vectors as Equation (2).

$$s = tok(S) + pos(S) + seg(S), \tag{2}$$

where $tok(S)$, $pos(S)$ and $seg(S)$ represent the token embedding, position embedding and segment embedding of sequence S, respectively. The parameters of token embedding and position embedding are initialized by the pre-trained model. In particular, the segment embedding of $S1$ is set to 0, and the segment embedding of the $S2$ is set to 1.

Transformer block: The bulk of the UniLM is made up of a stack of L Transformer blocks, each consisting of multi-headed self-attention and feed forward. The initial input vector is H^0 as Equation (3).

$$H^0 = \{s_i\}_{i=1}^{|s|} = \left[s_1, s_2, \ldots, s_{|s|}\right], \tag{3}$$

[2]https://github.com/bojone/BERT4keras

where $\{s_i\}_{i=1}^{|s|}$ denotes the input vectors, s_i represents the model input vector corresponding to the $i-th$ token of the sequence S, and $|s|$ denotes the number of input vectors. The context representation is then computed by the transformer block of the L layer. Each transformer block fuses the output of the previous layer using multi-headed self-attention, and after fitting the multilayer network in feed forward to get the output H^l of this layer as Equation (4). Multi-headed self-attention consists of multiple self-attention mapped from different spaces in parallel, and sums up the results of different self-attention extractions.

$$H^l = Transformer_l(H^{l-1}), l \in [1, L] \tag{4}$$

Masked attention: SOMPT uses Seq-to-Seq attention matrix $M_{Seq-to-Seq}$ to add prior constraints for attention mechanism which is implemented by UniLM. We mask sequence pairs (S_1, S_2) by using the Seq_to_seq mask matrix in the process of model training. Its corresponding mask matrix representation is shown in Figure 6 and the formula is represented as Equation (5), where j denotes the coordinate x, i denotes the coordinate y.

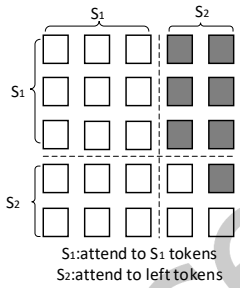

Fig. 6. Seq-to-Seq self-attention mask matrix.

$$\mathbf{M}_{ij} = \begin{cases} 0, \text{other.} \\ -\infty, i \in S_1, j \in S_2; i < j, i \in S_1, j \in S_2 \end{cases} \tag{5}$$

The output vector after the L-layer transformer block is denoted as H^{l-1}. After mapping the corresponding parameter matrices, the $\boldsymbol{Q} \in \mathbb{R}^{l_q \times d_q}, \boldsymbol{K} \in \mathbb{R}^{l_k \times d_q}, \boldsymbol{V} \in \mathbb{R}^{l_k \times d_v}$ matrices are gained as Equation (6). Each row of the $\boldsymbol{Q}$ matrix is a query vector, which is inner-producted with the key vector of each column in the $\boldsymbol{K}$ matrix to obtain the similarity. Then the normalized similarity is obtained using softMax after $\sqrt{d_k}$ adjustment, and the vector $\boldsymbol{V}$ is weighted and summed according to this similarity to acquire a self-attention head $\mathbf{A}_l$ as Equation (7). The Attention used by SOMPT adds the $\mathbf{M}_{\text{Seq-to-Seq}}$ matrix relative to the Seq-to-Seq LM, which is summed with $\mathbf{QK}^T$ to achieve masking of the Attention matrix.

$$Q = H^{(l-1)}W_l^Q, K = H^{(l-1)}W_l^K, V = H^{(l-1)}W_l^V \tag{6}$$

$$A_l = \text{softMax}\left(\frac{QK^T}{\sqrt{d_k}} + M_{\text{Seq-to-Seq}}\right)V \tag{7}$$

Knowledge point tagging: SOMPT model is fine-tuned by masking all tokens in the target sequence what we call knowledge points, and learning to recover the masked words. SOMPT's training objective is to maximize the likelihood of masked knowledge points tokens given question text. During training, we use the teacher-forcing mechanism, which has become the main training paradigm for sequence generation models [12, 14, 39]. Specifically, the tokens generated by the model will be replaced by true tokens and then participate in the next step of training, reducing the instability and difficulty of fitting the model caused by prediction errors during training. Finally, we get a vector $Z = (z_1, z_2, \ldots, z_{|V|})$ of size $|V|$, which is the size of the word list D. Subsequently,

softmax is performed on D to obtain the knowledge points probability matrix. The probability is calculated as Equation (8).

$$p_i = \frac{e^{z_i}}{\sum_i e^{z_i}} \tag{8}$$

According to this probability matrix, we query the word segmentation list to get the words of the generated sequence, and split it by the split function in python library with commas to get the knowledge points set $W = \{w_1, w_2, \ldots, w_n\}$.

What needs to be emphasized is that when tagging knowledge points for one question, SOMPT will use [MASK] as the initial token of S_2. [MASK] is an alternative to the real token of the sequence and is used to keep the model from knowing the predicted result in advance. The input sequence pair of the model is ([CLS], X, [SEP], [MASK],..., [MASK], [SEP]), which is different from the data input during model training. And then repeatedly adding the generated target token to the end of the source sequence to replace the previous [MASK] token during the prediction process, and stopping when [SEP] is encountered. Through the attention mechanism, this [MASK] can obtain the information of the sequence ([CLS], X, [SEP], [MASK]) and then predict the word vector representation.

3.5 Loss Function

After obtaining the knowledge point probability matrix, we need to calculate the error between the predicted labels and the true labels. Thus, we use the cross-entropy loss to measure the loss between the predicted probability and the true value to optimize the parameters of the SOMPT model. We tune the model parameters by minimizing the loss so that the model can be trained to the best tagging state. The formula is as Equation (9) and Equation (10),

$$\mathcal{L}(Q) = -\sum_{i=1}^{n} \left(I\left(w_i\right) \log\left(v_i\right) + \left(1 - I\left(w_i\right)\right) \log\left(1 - v_i\right)\right) \tag{9}$$

$$I\left(w_i\right) = \begin{cases} 1, & w_i \in L' \\ 0, & w_i \notin L' \end{cases} \tag{10}$$

where $I(w_i)$ denotes the probability that the knowledge point w_i belongs to the question, n denotes the number of knowledge points per question. Supposing the probability that each knowledge point predicted by the model belongs to a question is v_i, the predicted knowledge point of the question is w_i.

4 EXPERIMENTS

4.1 Knowledge point tagging experiments in Mathematical

4.1.1 ***Datasets***. TMK-PSS: Standard dataset for tagging knowledge points for math questions in primary and secondary schools. At present, there are few related researches on subject questions knowledge point tagging, and no suitable open datasets have been found. TMK-PSS used in the experiment comes from a learning big data platform from Central China Normal University in China[3]. The dataset is in Chinese. All questions are uploaded by the teacher, and the uploaded information includes question ID, knowledge points, question stem, question resolve, difficulty value, subject category, applicable grade, etc. Each question is marked by the subject teacher with multiple multilevel labels. TMK-PSS data set consists of non-repetitive questions, including the stem and resolve of questions, and selects all the No. 1 level labels as knowledge points. Figure 1 shows an example of data in TMK-PSS. The specific statistics are shown in Table 2. We randomly divide the dataset in the ratio of 6:2:2 to get the training set, validation set, and test set for experiments.

[3]http://study.hub.nercel.com/

Table 2. Details of the experimental dataset. N represents the dataset size. L means the total number of labels in the dataset. T denotes the average number of labels per sample. W represents the average Chinese characters or words of per sample.

Dataset	N	L	T	W
TMK-PSS	68522	97	1.22	158.6

4.1.2 ***Evaluation Metrics.*** For the performance of different models in the tagging task, we adopt hamming loss(HL) as our main evaluation metrics according to the previous work [40]. Accuracy, Precision, and Recall are also reported to assist the analysis. The definitions are as follows.

(1) HL is used to examine the misclassification of samples on a single label. The smaller the value of this index, the better the classification performance of the model. Its optimal value is 0, which means that all labels of each data are correctly classified and is calculated as Equation (11).

$$HL = \frac{1}{NM}\sum_{n=1}^{N}\sum_{m=1}^{M} I\left(y_m^n \neq w_m^n\right) \tag{11}$$

where y_m^n denotes the $m-th$ knowledge point of the $n-th$ sample, w_m^n denotes the $m-th$ label predicted by the $n-th$ sample, and $I(\cdot)$ is the indicator function, which takes 1 when y_m^n Is exactly equal to w_m^n, 0 otherwise.

(2) Accuracy is the ratio of the number of correctly identified samples in the total number of samples to be tested, which is a common measurement indicator and the most intuitive comparison method.

(3) Precision is the ratio of true positive samples among the positive cases determined by the classifier.

(4) Recall is the proportion of correctly determined positive cases to the total positive cases, indicating the average validity of each class for which the classifier identifies the class labels.

The formulas for Accuracy, Precision, and Recall are Equations (12)-(14).

$$Accuracy = \frac{1}{L}\sum_{i=1}^{N} m_i \frac{TP_i + TN_i}{TP_i + FN_i + FP_i + TN_i} \tag{12}$$

$$Precision = \frac{1}{L}\sum_{i=1}^{N} m_i \frac{TP_i}{TP_i + FP_i} \tag{13}$$

$$Recall = \frac{1}{L}\sum_{i=1}^{N} m_i \frac{TP_i}{TP_i + FN_i} \tag{14}$$

where L denotes the total number of samples, N denotes the total number of classes, m_i denotes the number of samples of the $i-th$ class. Suppose TP, FP, TN, and FN denote determined as category i and determined correctly, determined as category i but determined incorrectly, determined as not category i but determined incorrectly, and determined as not category i and determined correctly, respectively.

4.1.3 ***Experimental Setup.*** We use the pre-trained model of the base version of BERT to initialize the parameters of the input embedding and the transformer blocks of the SOMPT model. The embedding Layer is initialized using the BERT pre-training model, so the dimensionality of its embedding layer is 768. The labels in the generated sequences are represented by short text, so their word vectors must be very different from the pretraining word vectors. We do not set the weights as static, but dynamically update the corresponding weights during the training process. The number of transformer blocks (i.e., L) is set to 12, and the activation function is Gelu. The dropout is 0.1. Gradient descent is performed using Adam with a learning rate of 2e-5 as the optimizer. The batch_size is set to a multiple of 4, with a maximum of 32. We introduce a beam search mechanism in the sequence generation

part of the model. The model retains the current optimal result at each step of sequence generation. We repeat each experiment independently 5 times and record the average performance of the model.

4.1.4 ***Baselines.*** The baselines for experimental comparison with the SOMPT model are built based on deep learning models commonly used in the field of multi-label classification as follows.

(1) TextCNN[20, 51]: TextCNN is a type of convolutional neural network that utilizes different sizes of convolutional neural networks to continuously capture contextual connections.

(2) LSTM-ATT[44]: The LSTM [16] model is most commonly used for sequence generation of text and is an extension of RNN. Text classification is enhanced by adding the attention mechanism[41] to extract richer text features.

(3) BiLSTM-ATT[36, 53]: Bi-LSTM combines the information of the input sequence in both forward and backward directions on the basis of LSTM, which can capture more information of the text, and we add the attention mechanism to improve the model.

(4) BERT[4, 45]: We adopt the official Google pre-trained model parameters, using the [CLS] vector of BERT outputs as the sentence vector for textual multi-label prediction.

(5) UniLM[11]: We regard knowledge point labels as generated sequences, and use UniLM to complete the process of generating tag sequences from mathematical questions for the tagging task.

For the above five deep learning models, the sentence vector representation of their output is utilized, and we building a network for classification. The sentence vector is finally input into a fully connected network and normalized by softMax to get an output probability matrix. According to the predicted probability of each knowledge point in the probability matrix, multiple labels are derived for each question.

4.1.5 ***Results and analysis.*** We conduct specific mathematical question tagging experiments on baselines and the proposed model using TMK-PSS dataset. The experimental results are shown in Table 3.

Table 3. The tagging performance of the baselines and SOMPT. The (+) represents the higher score the better performance, and the (-) is the opposite.

Model	TMK-PSS			
	HL(-)	Precision(+)	Recall(+)	Accuracy(+)
TextCNN	0.00884	0.715	0.485	0.426
BiLSTM-ATT	0.00954	0.646	0.521	0.426
LSTM-ATT	0.00964	0.633	0.541	0.444
BERT	0.00879	0.687	0.556	0.496
UniLM	0.00655	0.742	0.736	0.668
SOMPT	**0.00631**	**0.762**	**0.725**	**0.678**

In general, SOMPT that our proposed performs significantly better than the baselines model in the standard dataset of TMK-PSS. In the following, we will analyze the experimental results in detail. Firstly, the deep learning models used for comparison, the TextCNN model, performs better than the other two RNN models, mainly because the TextCNN model captures several different n-gram features of text. It can extract useful information from different perspectives for the same n-gram feature. However, the Precision and Recall of TextCNN model differ greatly, indicating that the model is conservative when tagging and can only predict the knowledge points that are highly correlated with the question. Moreover, for the knowledge point tagging task, the performance of BiLSTM-ATT among the RNN series models is better, and the performance of the LSTM-ATT model is slightly worse. Since the contextual connection of math question text is not strong enough, RNN series models capture less contextual connection, which makes the effect will have a certain gap with TextCNN model which is more

focused on extracting shallow features of text. As the text of mathematical questions is not short, the BiLSTM-ATT model can take advantage of its suitability for long text and stable performance in our dataset and makes its experimental results slightly better than LSTM-ATT model. Next, the experimental results of BERT are a little better compared with the above models , because the BERT model is pre-trained and fine-tuned to produce a very rich contextual representation, and tagging for questions relies on the model's ability to extract and analyze the semantic information of the question text. Compared to our model SOMPT, the experimental results of BERT are inferior. Although our model has a similar framework to BERT in extracting semantic information, the subsequent classification is handled differently. This indicates that it is not enough to consider only the semantic information but not the connection between knowledge points when tagging.

Obviously, our SOMPT model shows the best performance on the TMK-PSS. Compared with BERT, which performs relatively well, the SOMPT model is 7.5%, 16.9%, and 18.2% higher in the evaluation metrics Precision, Recall, and Accuracy, respectively, and 0.00248 lower on HL. Even compared to UniLM, the SOMPT model has a significant improvement. The results prove to some extent the feasibility and correctness of using the idea of sequence generation to accomplish the task of automatic tagging multi-knowledge points. What is more exciting is that the Precision and Recall values of SOMPT model are almost the same, which indicates that the classification performance of SOMPT model is stronger and more stable when tagging knowledge points for mathematical questions.

We plot the convergence of the six tagging methods of mathematical questions, as shown in Figure 7. As we can see, except for BiLSTM-ATT, the fluctuations of other models are relatively small. Among the six models, SOMPT model has the fastest Loss convergence and the most stable convergence curve. After 10 rounds of training, the model will reach a close convergence state. BERT is gradually approaching a state of convergence after almost 20 rounds. During the early stages of training, the weights of the model have not been sufficiently adjusted to accurately fit the training data. As a result, both BERT and LSTM display similar loss values, while SOMPT exhibits superior performance from the outset. These all indicates that SOMPT model has a stronger ability to capture data features and adjust model parameters more quickly when training data.

4.2 Ablation experiments of SOMPT

To explore whether data augmentation based on ontology replacement (The part (a) of SOMPT) for question text can improve the performance knowledge point tagging, we perform ablation experiments for the ontology-based replacement module. Table 4 shows the experimental results of question tagging on TMK-PSS that baselines are added ontology-based replacement module and SOMPT is removed ontology-based replacement module. By observing the last two lines of Table 6, it can be found that HL increases and Accuracy decreases after the ontology-based replacement module is removed, indicating that SOMPT's tagging performance for mathematical questions is reduced. In addition, comparing the performance of baselines in Table 3 and Table 4, we can find that the experimental result of baseline is improved after the addition of ontology-based replacement module. It proves that the ontology-based replacement on the dataset is necessary. And it also shows that the method based on integrating domain ontology and named entity replacement enables the knowledge point tagging model to dig deeper into the text information that is beneficial to tagging performances.

In order to further explore the impact of the data augmentation method on mathematical question representation, we selected a sample for a case study. This sample involves two knowledge points of "recognition of large numbers" and "recognition of integers". We use SOMPT-(a) and SOMPT to extract representations of sample question and knowledge points respectively, and then we use Euclidean distance and cosine similarity to calculate the distance between question and label representations, as shown in Table 5. The smaller the Euclidean distance, the greater the cosine similarity, indicating that the closer the distance between the two, the more similar they are. We can see that compared with SOMPT-(a), SOMPT has achieved significant improvements

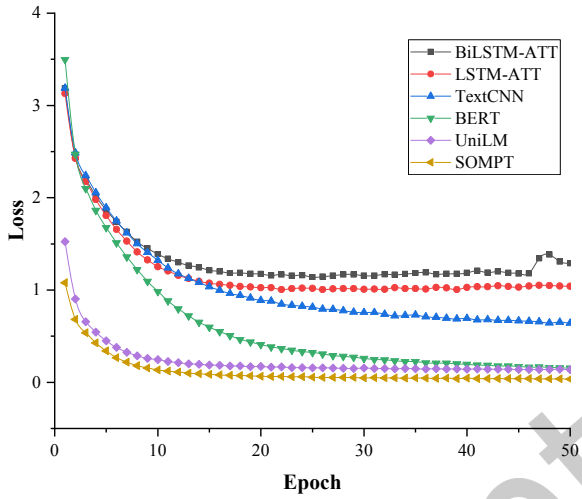

Fig. 7. Loss convergence fluctuation diagram during model training.

Table 4. Ablation experiments on the ontology-base replacement module of SOMPT. "+SOMPT(a)" represents the model is added the part (a) of SOMPT. "SOMPT-(a)" represents the model SOMPT is removed the part (a). The (+) represents the higher score the better performance, and the (-) is the opposite.

Model	Metrics			
	HL(-)	Precision(+)	Recall(+)	Accuracy(+)
TextCNN+SOMPT(a)	0.00828	0.752	0.512	0.456
BiLSTM-ATT+SOMPT(a)	0.00891	0.669	0.565	0.460
LSTM-ATT+SOMPT(a)	0.00922	0.650	0.583	0.465
BERT+SOMPT(a)	0.00815	0.707	0.604	0.533
SOMPT-(a)	0.00655	0.742	0.736	0.668
SOMPT	**0.00631**	**0.762**	**0.725**	**0.678**

in both Euclidean distance and cosine similarity, which illustrates the effectiveness of our data augmentation method. In the knowledge point tagging task, the data augmentation method helps the model learn more useful features related to knowledge points by removing redundant information and focusing on key information.

4.3 Applications on multi-label classification task

The module used for knowledge point tagging (module (b) and (c) in Figure 2, code-named SOMPT(b)(c)) in SOMPT model proposed in this paper can also be applied to MLC task. In order to verify the effect on multi-label classification task, we select two MLC publicly datasets AAPD[48] and RCV1-V2[23] for extension experiments. The AAPD dataset includes abstracts and topics of academic papers in the field of computer science, while RCV1-V2 collects news texts and related topics. We take a sample in AAPD as an example, the text is "we give a characterization of vertex monotone properties with sharp thresholds in a poisson random geometric graph or

Table 5. Euclidean distance and cosine similarity between question and label representations. Label 1 means "knowledge of large numbers", and label 2 means "knowledge of integers".The (+) represents the higher score the more similar, and the (-) is the opposite.

Model	SOMPT-(a)		SOMPT	
Metrics	Distance (-)	Cosine (+)	Distance (-)	Cosine (+)
Label 1	6.940	0.840	5.661	0.897
Label 2	7.488	0.814	6.180	0.877

hypergraph as an application we show that a geometric model of random k sat exhibits a sharp threshold for satisfiability", and the labels for the topics are "[30, 18, 32, 28]". These two datasets are in English and are not a problem in the mathematical domain, so we did not preprocess the dataset text with ontology-based replacement. We also applied a short text replacement module (module b) to both datasets. We first count all the words that appear in the text, and then select unused words from the dictionary to replace the digital labels of the topic labels. Table 6 shows the experimental results of our proposed model, the baseline of this paper, and representative models from other papers.

Table 6. Performance on AAPD and RCV1-V2. The ones with † are the experimental results obtained directly from others' papers.

Dataset	AAPD				RCV1-V2			
Metrics	HL (-)	Precision (+)	Recall (+)	Accuracy (+)	HL (-)	Precision (+)	Recall (+)	Accuracy (+)
BR† [3]	0.0316	0.664	0.648	-	0.0086	0.904	0.816	-
Seq-ATT†[37]	0.0261	0.720	0.639	-	0.0081	0.889	0.848	-
SGM† [48]	**0.0245**	**0.748**	0.675	-	**0.0075**	**0.897**	**0.860**	-
TextCNN	0.0290	0.741	0.536	0.295	0.0134	0.852	0.686	0.460
BiLSTM-ATT	0.0307	0.716	0.513	0.287	0.0141	0.834	0.679	0.447
LSTM-ATT	0.0305	0.683	0.585	0.300	0.0135	0.836	0.700	0.462
BERT	0.0304	0.703	0.549	0.319	0.0144	0.802	0.709	0.485
SOMPT(b)(c)	**0.0258**	**0.718**	**0.690**	**0.401**	0.0095	0.875	0.834	0.586

The experiment on AAPD. The HL of SOMPT(b)(c) decreases by 0.0058 compared to the most commonly used model BR [18], and Precision and Recall improve by nearly 5%. In addition, compared to SGM [1], the most advanced model in the field of MLC, the numerical difference of all evaluation indicators is very small. Compared with other models that have been proposed, SOMPT(b)(c) model has the best performance on major evaluation metrics, only lower than Seq2seq attention [27] on Precision. Considering that Precision and Recall of good classification models need to be balanced, therefore, the performance of SOMPT(b)(c) model is relatively better. There is no doubt that SOMPT(b)(c) performs better on all evaluation metrics than the baseline used in this paper.

The experiment on RCV1-V2. In contrast to the experimental results on the AAPD dataset, our proposed method outperforms all this paper baselines on evaluation metrics, but it is not as good as the results of the model in other's paper. Considering that the improvement of MLC performance on RCV1-V2 by researchers in recent years is limited and not as great as the AAPD data set, it shows that it is still difficult to explore a model with perfect performance on RCV1-V2. In addition, the HL of SOMPT(b)(c) can reach 0.0095 in RCV1-V2, which is just 0.002 higher than SGM. This also indicates that SOMPT(b)(c), which is suitable for multi-knowledge point

tagging of mathematical questions, is similarly effective on the MLC task, but it is not applicable to all multi-label text classification standard datasets.

5 CONCLUSIONS

With the popularity of educational recommendation tools, the volume of task and the difficulty of tagging learning resource has increased, putting forward higher requirements on the automation techniques of tagging tasks. Aiming at the task of mathematical question tagging, we propose a novel sequence generation model integrating domain ontology. By pre-processing text with ontology replacement to eliminate the negative impact of many entities and different forms of expression in text on the deep learning model. Then we vectorized the intermediate text and fed into the question tagging model based on sequence generation to tagging knowledge points for mathematical questions. Our proposed model has the ability to capture the rich semantic information of the question, and to tag the questions in the specific mathematical domain in a new way of sequence generation, which can catch the relationship between knowledge points of the question. Moreover, the proposed model not only performs well on the multi-knowledge point tagging dataset, but also has good experimental results on two publicly available multi-label classification standard datasets. In the process of exploring personalized recommendation technology in the future, we will continue to expand our dataset, construct more specialized multi-knowledge point tagging datasets, and put forward multi-knowledge point tagging models with higher performance.

ACKNOWLEDGMENTS

This research was supported by the National Natural Science Foundation of China (Grant No.61977033, No.62077024), the State Key Program of National Natural Science of China (Grant No.U20A20229), and the Fundamental Research Funds for the Central Universities (Grant No.CCNU20TD007, No.CCNUTEIII 2021-03).

REFERENCES

[1] Peter Babinec and Ivan Srba. 2017. Education-specific tag recommendation in CQA systems. In *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*. 281–286.

[2] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018).

[3] Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. 2004. Learning multi-label scene classification. *Pattern recognition* 37, 9 (2004), 1757–1771.

[4] Linkun Cai, Yu Song, Tao Liu, and Kunli Zhang. 2020. A hybrid BERT model that incorporates label semantics via adjustive attention for multi-label text classification. *Ieee Access* 8 (2020), 152183–152192.

[5] Suleyman Cetintas, Luo Si, Yan Ping Xin, Dake Zhang, Joo Young Park, and Ron Tzur. 2014. A joint probabilistic classification model of relevant and irrelevant sentences in mathematical word problems. *Journal of Educaltional Data Mining* 2, 1 (2014), 83–101.

[6] Guibin Chen, Deheng Ye, Zhenchang Xing, Jieshan Chen, and Erik Cambria. 2017. Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In *2017 International joint conference on neural networks (IJCNN)*, Vol. 2017-May. IEEE, 2377–2383.

[7] Penghe Chen, Yu Lu, Vincent W Zheng, Xiyang Chen, and Boda Yang. 2018. Knowedu: A system to construct knowledge graph for education. *Ieee Access* 6 (2018), 31553–31563.

[8] Xiaohong Chen, Huanhuan Chen, Zhijia Fang, Tong Ruan, and Haofen Wang. 2017. Research And Implementation Of Annotation Algorithm For Walkthrough Text Based On Domain Ontology. *Computer Applications and Software* 34, 2 (2017), 80–86.

[9] Yewang Chen, Wen Li, Xin Peng, and Zhao Wenyun. 2009. Improved semantic annotation method for documents based on ontology. *Journal of Southeast University (Natural Science Edition)* 39, 6 (2009), 1109–1113.

[10] Guo Chonghui and LV Zhengda. 2020. Chinese Medicine Data Process Platform Based on Semantic Annotation. *OPERATIONS RESEARCH AND MANAGEMENT SCIENCE* 29, 2 (2020), 129–136.

[11] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Advances in Neural Information Processing Systems* 32 (2019).

[12] Konstantinos Drossos, Shayan Gharib, Paul Magron, and Tuomas Virtanen. 2019. Language modelling for sound event detection with teacher forcing and scheduled sampling. *arXiv preprint arXiv:1907.08506* (2019).
[13] Xue Fei. 2021. An LDA based model for semantic annotation of Web English educational resources. *Journal of Intelligent & Fuzzy Systems* 40, 2 (2021), 3445–3454.
[14] Yang Feng, Shuhao Gu, Dengji Guo, Zhengxin Yang, and Chenze Shao. 2021. Guiding Teacher Forcing with Seer Forcing for Neural Machine Translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2862–2872.
[15] Chonghui Guo, Xiaoyu Xing, and Wei Wei. 2021. A Knowledge Points Labeling Method for Test Questions Based on Bipartite Graph. *OPERATIONS RESEARCH AND MANAGEMENT SCIENCE* 30, 11 (2021), 2–7.
[16] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[17] Guoping Hu, Dan Zhang, Yu Su, Jia Li, Qingwen Liu, and Rui Wang. 2018. Predicting Knowledge Points of Questions: an Expertise Enriched CNN Model. *Journal of information Processing* 32, 5 (2018), 137–146.
[18] Yujia Huo, Derek F Wong, Lionel M Ni, Lidia S Chao, and Jing Zhang. 2020. Knowledge modeling via contextualized representations for LSTM-based personalized exercise recommendation. *Information Sciences* 523 (2020), 266–278.
[19] Kenton Lee Jacob Devlin, Ming-Wei Chang and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*. 4171–4186.
[20] Y. Kim. 2014. Convolutional Neural Networks for Sentence Classification. *EMNLP 2014-2014 Conf.Empir.Methods Nat. Lang. Process. Proc. Conf.* (2014), 1746—-1751.
[21] Phusavat Kongkiti, Yang Harrison Hao, et al. 2021. Shaping the future learning environments with smart elements: challenges and opportunities. *International Journal of Educational Technology in Higher Education* 18, 1 (2021), 1–9.
[22] Yakun Lang and Guozhong Wang. 2020. Personalized knowledge point recommendation system based on course knowledge graph. In *Journal of Physics: Conference Series*, Vol. 1634. IOP Publishing, 012073.
[23] David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research* 5, Apr (2004), 361–397.
[24] Weizhi Liao, Yu Wang, Yanchao Yin, Xiaobing Zhang, and Pan Ma. 2020. Improved sequence generation model for multi-label classification via CNN and initialized fully connection. *Neurocomputing* 382 (2020), 188–195.
[25] Junyang Lin, Qi Su, Pengcheng Yang, Shuming Ma, and Xu Sun. 2018. Semantic-unit-based dilated convolution for multi-label text classification. *Proc. 2018 Conf. Empir. Methods Nat. Lang. Process. EMNLP 2018* (2018), 4554–4564.
[26] Jinjiao Lin, Yanze Zhao, Chunfang Liu, and Haitao Pu. 2020. Personalized learning service based on big data for education. In *2020 IEEE 2nd International Conference on Computer Science and Educational Informatization (CSEI)*. IEEE, 235–238.
[27] Weijun Lu, Yun Duan, and Yutong Song. 2020. Self-Attention-Based Convolutional Neural Networks for Sentence Classification. In *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*. IEEE, 2065–2069.
[28] Setareh Maghsudi, Andrew Lan, Jie Xu, and Mihaela van Der Schaar. 2021. Personalized education in the artificial intelligence era: what to expect next. *IEEE Signal Processing Magazine* 38, 3 (2021), 37–50.
[29] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *1st Int. Conf. Learn. Represent. ICLR 2013 - Work. Track Proc.* (2013), 1–12.
[30] Jinseok Nam, Eneldo Loza Mencía, Hyunwoo J Kim, and Johannes Fürnkranz. 2017. Maximizing subset accuracy with recurrent neural networks in multi-label classification. *Advances in neural information processing systems* 30 (2017), 5414–5424.
[31] Zhenglin Ni and Fangwei Ni. 2020. Research on knowledge graph model of diversified online resources and personalized recommendation. In *Journal of Physics: Conference Series*, Vol. 1693. IOP Publishing, 1–7.
[32] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. *Advances in neural information processing systems* 28 (2015), 505––513.
[33] Kechen Qin, Cheng Li, Virgil Pavlu, and Javed A Aslam. 2019. Adapting RNN sequence prediction model to multi-label set prediction. *NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.* 1 (2019), 3181–3190.
[34] Alexandru Stefan Stoica, Stella Heras, Javier Palanca, Vicente Julián, and Marian Cristian Mihaescu. 2021. Classification of educational videos by using a semi-supervised learning method on transcripts and keywords. *Neurocomputing* 456 (2021), 637–647.
[35] Bo Sun, Yunzong Zhu, Yongkang Xiao, Rong Xiao, and Yungang Wei. 2018. Automatic question tagging with deep neural networks. *IEEE Transactions on Learning Technologies* 12, 1 (2018), 29–43.
[36] Bo Sun, Yunzong Zhu, Zeng Yao, Rong Xiao, Yongkang Xiao, and Yungang Wei. 2020. Tagging Reading Comprehension Materials With Document Extraction Attention Networks. *IEEE Transactions on Learning Technologies* 13, 3 (2020), 567–579.
[37] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems* 27 (2014).
[38] Tokinori Suzuki and Atsushi Fujii. 2017. Mathematical document categorization with structure of mathematical expressions. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. IEEE, 1–10.

[39] Nikzad Benny Toomarian and Jacob Barhen. 1992. Learning a trajectory using adjoint functions and teacher forcing. *Neural networks* 5, 3 (1992), 473–484.
[40] Grigorios Tsoumakas and Ioannis Katakis. 2007. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)* 3, 3 (2007), 1–13.
[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
[42] Z.Y. Shao W. Wang and J.Y. Zhou. 2017. A personalized exercises recommendation system based on knowledgepoints and its application in"basic of medical computer application. *Zhejiang Medical Education* 19, 4 (2017), 4–7.
[43] Peng Wang, Jiaming Xu, Bo Xu, Chenglin Liu, Heng Zhang, Fangyuan Wang, and Hongwei Hao. 2015. Semantic clustering and convolutional neural network for short text categorization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. 352–357.
[44] S.K. Wang. 2019. Knowledge Point Marking System Based on LSTM and Attention, Vol. University of Electronic Science and Technology.
[45] Zhongju Wang, Long Wang, Chao Huang, and Xiong Luo. 2021. BERT-based Chinese Text Classification for Emergency Domain with a Novel Loss Function. *arXiv preprint arXiv:2104.04197* (2021).
[46] Xing Xiaoyu Wei Wei, Guo Chonghui. 2020. Annotating Knowledge Points & Recommending Questions Based on Semantic Association Rules. *Data Analysis and Knowledge Discovery* 4, 2/3 (2020), 182–191.
[47] Guoqiang Wu, Ruobing Zheng, Yingjie Tian, and Dalian Liu. 2020. Joint ranking SVM and binary relevance with robust low-rank learning for multi-label classification. *Neural Networks* 122 (2020), 24–39.
[48] Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: sequence generation model for multi-label classification. *arXiv preprint arXiv:1806.04822* (2018), 3915–3926.
[49] Chen Yanjun and Li Keda. 2020. Chinese Medicine Data Process Platform Based on Semantic Annotation. *Computer Applications* 39, 9 (2020), 37–40.
[50] Fu Z. 2016. A Review of Semantic Annotation. *Research on Library Science* 2016, 4 (2016), 10–17.
[51] Qiang Zhang, Rongrong Zheng, Ziyan Zhao, Bo Chai, and Jiangui Li. 2020. A textcnn based approach for multi-label text classification of power fault data. In *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*. IEEE, 179–183.
[52] XiuQin Zhong, HongGuang Fu, She Li, and Huang Bin. 2010. Geometry Knowledge Acquisition and Representation on Ontology. *CHINESE JOURNAL OF COMPUTERS* 33, 1 (2010), 167–174.
[53] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th annual meeting of the association for computational linguistics*, Vol. 2(Short papers). 207–212.
[54] TY Zhu, Zhenya Huang, Enhong Chen, Qi Liu, Runze Wu, Le Wu, and Guoping Hu. 2017. Cognitive diagnosis based personalized question recommendation. *Chinese Journal of Computers* 40, 1 (2017), 176–191.