

基于对比学习的数学应用题求解方法研究

张天成, 王玉杨, 张亦嘉, 于明鹤, 冷芳玲, 于戈
东北大学计算机科学与工程学院, 辽宁 沈阳 110819

摘要

MWP自动求解不仅可以为学生提供精准的学习指导, 还有助于缓解教师压力, 推动智慧教育发展。现有MWP求解方法在挖掘题目深层语义信息、区分MWP类型以及处理相似问题的细微差异方面仍存在挑战。为此, 提出了一种通用的数学应用题求解模型BSCL。首先, 通过预训练语言模型对自然语言形式的MWP进行编码, 同时利用对比学习方法增强编码器对不同题型的理解能力; 然后, 统一解码多种类型的MWP, 并采用监督任务确保题目与表达式之间的数学一致性。在中文和英文数据集上的多项实验结果证明了BSCL求解不同类型MWP的有效性和优越性。

关键词

数学应用题; 自然语言处理; 对比学习

中图分类号: TP399

文献标志码: A

doi:10.11959/j.issn.2096-0271.2025057

Study on solving math word problem based on contrastive learning

ZHANG Tiancheng, WANG Yuyang, ZHANG Yijia, YU Minghe, LENG Fangling, YU Ge
School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

Abstract

The automatic solving of MWP not only provides students with accurate learning guidance but also helps alleviate teacher workload and promotes advancements of smart education. Existing MWP-solving methods face challenges in extracting deep semantic information and unifying the solutions for various types of MWP, and addressing the subtle differences in similar problems. To address these challenges, a general MWP solving model, BSCL, was proposed. Firstly, pre-trained language models were used to encode MWP in natural language form, employing contrastive learning methods to enhance the encoder's ability to understand different MWP types. Then, the model unified the decoding of various types of MWP and used a supervision task to ensure mathematical consistency between the problems and expressions. Extensive experiments on both Chinese and English datasets indicate that BSCL has the effectiveness and superiority in solving different types of MWP tasks.

Key words

math word problem, natural language processing, contrastive learning

0 引言

数学应用题 (math word problem, MWP) 求解模型的实现是自然语言处理领域中的一项具有挑战性的任务, 其目标是自动求解自然语言描述的 MWP。该任务需要先通过编码模块将问题的自然语言描述转换为机器可理解的逻辑表示, 再通过解码模块将编码问题得到的嵌入表示映射为相应的数学表达式。根据未知数的数量, MWP 可分为单未知数的算术应用题 (arithmetic word problem, AWP) 和多未知数的方程组问题 (equation set problem, ESP) ^[1]。表 1 中例题 2 是一道算术应用题。在求解过程中, 模型首先从输入问题中提取出相关数值 800、30、5, 然后分析数量关系并推导出相应的运算符, 最后将问题映射为表达式 $x=800-30-5$, 并计算出最终答案为 765。

最早自动求解 MWP 的方法是基于规则的, 将问题转化为命题集, 再进行推理获得答案。基于统计机器学习的方法则使用传统机器学习模型识别未知数、运算符

和数值, 并通过逻辑推理得出表达式。但这些方法依赖预定义模板, 难以适应多样化的数据。随着深度学习的发展, 基于神经网络的端到端求解模型已被用于求解 MWP。这些模型大多用于解决单未知数的 AWP, 无法直接用于求解多未知数的 ESP。为此, SAU-Solver^[2]设计了一种通用表达式树 (universal expression tree, UET), 实现了不同类型 MWP 的统一求解方法。然而, 大多数求解模型容易误判 MWP 的类型。例如, 表 1 所示的例题 1 是一道 ESP, 但 SAU-Solver 却将其错误地识别为 AWP, 导致无法正确求解该问题。此外, 当前的求解模型还会忽略自然语言中蕴含的深层数学逻辑, 无法处理题目描述相似的 MWP 之间的细微差别, 从而无法推导出正确的求解表达式, 影响求解的准确性。如表 1 中例题 2、例题 3 所示, 两道 AWP 虽然表述相似, 但蕴含的数学逻辑却大不相同, 模型 GTS^[3]未能准确识别出这两道题目间的差异, 没有推断出例题 3 中数值 30 和数值 5 之间的乘法关系, 从而未能生成正确的数学表达式。

为了解决上述问题, 本文提出了一种基于对比学习的通用 MWP 求解模型 BSCL

表1 MWP样例

例题	求解过程
例题 1: ESP	一个两位数, 其各位数字之和是 7。当该两位数减去 27 时, 其十位和个位上的数字交换了位置。请问该两位数的十位数字和个位数字分别是多少?
正确表达式	$\begin{cases} x+y=7 \\ 10y+x-27=10x+y \end{cases}$
SAU-Solver 生成的表达式	$10x+x+2.0=10x+x$
例题 2: AWP	一本书共有 800 页, 明明上午读了 30 页, 下午读了 5 页, 还剩多少页没有读完?
正确表达式	$x=800-30-5$
GTS 生产的表达式	$x=800-30-5$
例题 3: AWP	一本书共有 800 页, 明明平均每小时读 30 页。他已经连续读了 5 小时, 还剩多少页没有读完?
正确表达式	$x=800-30 \times 5$
GTS 生成的表达式	$x=800-30-5$

(BERT-SAUE with contrastive learning)。考虑到预训练语言模型在自然语言理解中的优势，本文将其作为编码器，用于获取MWP的上下文表示。同时，本文结合一个基于多头注意力的监督模块^[4]，通过分类器来监督生成的表达式与问题文本之间的一致性，确保模型生成的表达式更加精准。此外，本文引入对比学习方法，通过对比损失函数优化编码器的问题表示，即在最大化原型方程相同MWP表示间相似度的同时，最小化不同类型MWP表示间的相似度，增强编码器对原型方程相同MWP表示间相似性和不同类型MWP表示间差异性的感知能力，从而提高模型对多种类型MWP的求解效果。最后，本文通过SAU-Solver实现对不同类型MWP的统一解码。本文主要贡献如下。

- 本文引入监督模块，确保MWP与数学表达式之间的一致性，使模型能区分文本描述相似但表达式不同的MWP，增强模型对MWP的理解能力。

- 本文提出一种对比学习方法，帮助模型理解不同类型MWP之间的相似性和差异性，增强模型对不同类型MWP的泛化性能。

- 本文在中文和英文数据集上进行了充分的测试。实验结果表明，BSCL在处理不同类型MWP时具有高效性和通用性。

1 相关工作

1.1 MWP自动求解

MWP求解在自然语言处理领域是一个非常具有挑战性的任务。2017年，Wang等^[5]提出了第一个基于深度神经网络的求解模型，与传统方法相比，该模型通过数据驱动的方式学习问题特征，节约了大量人工

成本。随后，基于Seq2Seq结构的多种深度学习求解模型相继被提出^[6-8]，这些模型将问题编码为词序列，但此类基于序列的方法仅关注词的序列影响，缺乏对问题的深层理解，基于序列的解码模型无法有效控制表达式推导，容易产生错误解答。为了解决这些问题，Xie等^[3]尝试通过将表达式生成步骤转化为构建UET来提升模型的推理能力。Zhang等^[9]通过图的形式编码数值比较关系和问题依赖关系，利用图神经网络学习问题表示。Zhang等^[10]设计了一种基于教师-学生网络的信息引导机制，用于表达式生成。Wu等^[11]从外部知识库中引入常识知识以构建实体图，并使用图神经网络学习知识表示。Qin等^[2]提出了通用表达式树UET，用统一的方式表示AWP和ESP。

Lin等^[12]提出了一种分层求解模型HMS，通过模仿人类阅读习惯，提出层次化的词-子句-问题编码器，并利用问题的依赖结构增强子句语义，同时在解码器中应用层次化注意力机制，通过来自不同层次的上下文增强问题语义。Lan等^[13]开发了第一个开源MWP求解库，集成多个深度学习模型和数据集，提供统一、全面且可扩展的框架。Zhou等^[14]基于启发式规则为每个训练样本生成一组与原样本相似但求解目标不同的新训练样本，扩充训练数据，突破了以往仅依赖浅层启发式方法求解MWP的局限性。Liang等^[15]提出一种组合型数学应用题求解器C-MWP，能够同时学习MWP涉及的语义模式和数量模式，理解文本的语义含义以及数量间的数学逻辑。Liang等^[16]提出了一种新的MWP求解模型训练框架，通过引入解答缓冲区和解答判别器，扩充并控制训练数据中的多样化解答，提升模型的泛化能力。Li等^[17]提出了一种新的数学方程相似度评估公式，

用于评估在 MWP 关联背景下数学方程的相似性，并探究了不同正负样本选择策略对对比学习模型的影响。

1.2 对比学习

对比学习是一种表示学习方法，最早由 Hadsell 等^[18]提出。其目的是使语义相似的语句在特征空间中彼此靠近，使语义不同的语句彼此远离，其自提出以来，在自然语言处理领域中得到了广泛应用。在机器翻译领域，Yang 等^[19]将省略若干词的句子作为翻译过程中的负样本，使用对比学习方法减少机器翻译中的词省略错误。Clark 等^[20]使用生成网络生成高质量的负样本，训练文本编码器以区分真实数据和负样本，从而获取更多的语言特征信息。Gao 等^[21]提出了一种对比学习框架，成功增强了句子嵌入在语义文本相似性任务中的表现性能，为文本输入的数据增强提供了一种新方法。Yu 等^[22]提出了一种对比正

则化自训练框架，通过促使同类数据具有相似表示，简化分类任务。

2 BSCL 模型介绍

2.1 模型总体框架

模型输入一道数学应用题 $Q = \{q_1, q_2, \dots, q_n\}$ ，其是一个长度为 n 的文本序列， q_i 为自然语言文本或数字。模型输出一个长度为 m 的数学表达式 $P_{eq} = \{p_1, p_2, \dots, p_m\}$ ，表达式中元素 p_i 可能来自出现在题目文本中的数字集合 V_{num} 、求解过程中需要用到外部辅助常数集合 V_{con} 、运算符集合 $V_{op} = \{+, -, *, /, ^, =, [SEP]\}$ 或变量集合 $V_{var} = \{x, y\}$ 。

BSCL 的总体架构如图 1 所示。首先，本文将预训练语言模型（pre-trained language model, PLM）作为编码器，编码自然语言形式的 MWP；然后，在训练

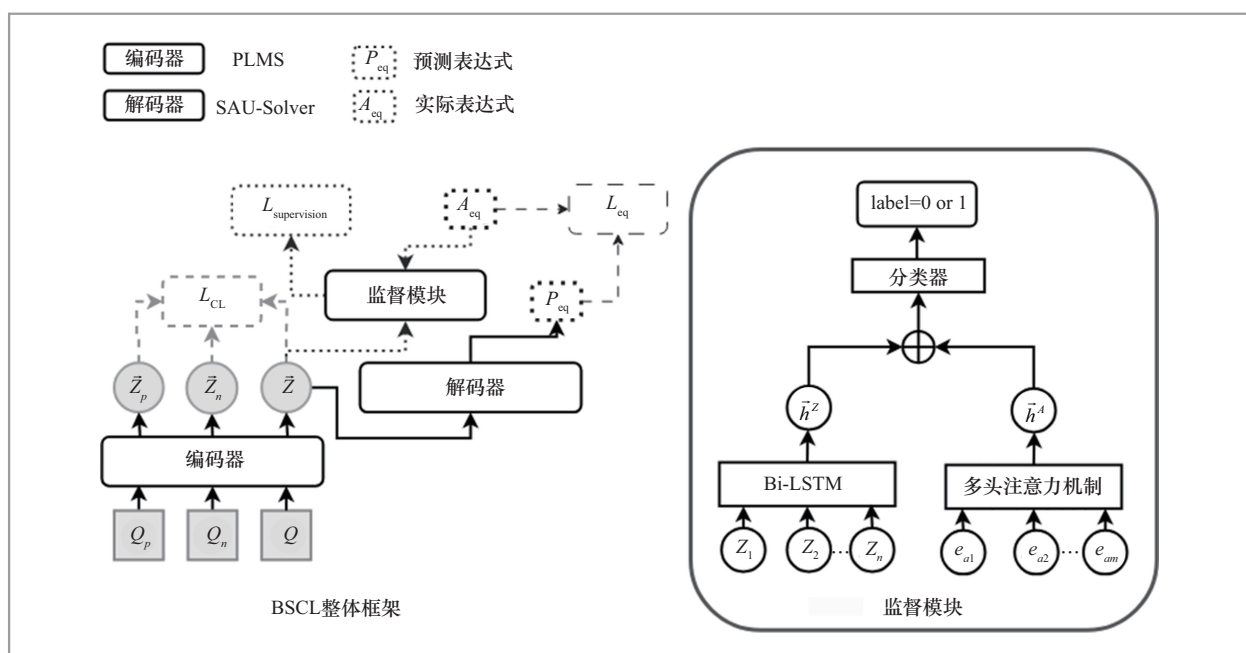


图1 BSCL 整体框架

过程中, 本文通过一致性检查监督任务和对比学习方法指导表示学习; 最后, 使用 SAU-Solver 提出的具有语义对齐正则化的树结构解码器, 解码生成每个问题对应的数学表达式。

2.2 模型构成

2.2.1 编码模块

PLM 在自然语言理解中具有显著优势。通过在大规模语料库上进行无监督预训练, PLM 学习丰富的语义和语法信息, 并捕捉语言的基本结构和模式。PLM 在小规模有标签数据集上进行微调, 从而更好地适应特定的下游任务, 实现对自然语言的深度表示。其中, BERT (bidirectional encoder representations from transformer) 由多个 Transformer 编码器堆叠而成, 其作为典型的 PLM, 通过两个自监督任务, 即采用掩码语言模型和下一个句子预测, 学习丰富的语义和上下文信息, 从而获取强大的语言表示能力。BERT 变体 RoBERTa^[23]则在更大数据集和更长训练时间的支持下进一步提升了下游任务的性能。

本文利用 RoBERTa 的语义表示能力实现对自然语言形式的数学应用题 $Q = \{q_1, q_2, \dots, q_n\}$ 的编码。首先, 编码模块计算每个元素 q_i 的嵌入向量 $p_i \in R^{1 \times 768}$, 该计算过程如式 (1) 至式 (4) 所示。

$$p_i^t = \text{WordPieces}_i W^t \quad (1)$$

$$p_i^s = \text{Segment}_i W^s \quad (2)$$

$$p_i^p = \text{Position}_i W^p \quad (3)$$

$$p_i = p_i^t + p_i^s + p_i^p \quad (4)$$

p_i^t 、 p_i^s 、 p_i^p 分别表示第 i 个子词的词嵌入向量、块嵌入向量和位置嵌入向量。 W^t 、 W^s 和 W^p 分别表示词嵌入矩阵、块嵌入矩阵和位置嵌入矩阵。 WordPieces_i 代表基于 WordPieces 分词方法得到的第 i 个子词, Segment_i 表示第 i 个子词属于哪一块, 由于在 MWP 求解任务中, 模型只输入一道应用题的文本描述, 在块嵌入过程中, 所有的子词均属于同一块, 默认为 0。 Position_i 表示第 i 个子词在文本序列中的绝对位置。

随后, 编码模块将应用题的嵌入表示 $P = \{p_1, p_2, \dots, p_n\}$ 输入多层 Transformer Encoder 模块, 获得应用题编码表示 $Z = \{z_1, z_2, \dots, z_n\}$ 。最后, 对 Z 进行平均池化操作, 得到应用题整体语义表示向量 $\vec{Z}$ 。

$$Z = \text{TransformerEncoder}(P) \quad (5)$$

$$\vec{Z} = \frac{1}{n} \sum_{i=1}^n z_i \quad (6)$$

2.2.2 解码模块

本文希望构建一个能够统一解码 AWP 和 ESP 的模型。因此, 本文采用 SAU-Solver 提出的通用表达式树 UET 结构, BSCL 通过树形解码器^[3]生成能够统一表示不同类型 MWP 的解表达式树。解码模块的训练目标是在给定 MWP 的文本描述 Q 的前提下, 最大化生成正确解表达式 P_{eq} 的概率。

UET 的每个节点包含 3 个核心部分: 目标向量 q 、数学符号 $\hat{y}$, 以及子树表示向量 t 。目标向量 q 用于预测节点对应的数学符号 $\hat{y}$, 通常用整体语义表示向量 $\vec{Z}$ 初始化。若数学符号 $\hat{y} \in \{V_{\text{Num}}, V_{\text{Con}}, V_{\text{vars}}\}$, 则子树分裂停止。若 $\hat{y} \in \{V_{\text{op}}\}$, 则开始以当前节点为根进行分裂, 并依次生成左右子

UET。当前节点的左右两棵子树均生成完毕后，树形解码器构建该节点的树嵌入向量 $\mathbf{t}$ ，并向上回溯直至整棵UET均构建完毕。图2是一道MWP的解表达式树的构建过程。树形解码器首先通过编码模块得到的该应用题整体语义表示向量 $\vec{Z}$ ，然后用 $\vec{Z}$ 来初始化表达式树的根节点的目标向量 $\mathbf{q}_{\text{root}}$ ，并将 $\mathbf{q}_{\text{root}}$ 压入栈（Stack）。接着，从栈中弹出一个节点的目标向量，树形解码器基于该目标向量预测当前节点对应的数学符号“/”，从而继续分裂生成左子节点的目标向量 $\mathbf{q}_l$ ，接着基于 $\mathbf{q}_l$ 预测“/”的左子节点为“+”，重复上述过程，直至树形解码器预测当前节点为数字“30”时停止分裂。随后，树形解码器生成以左节点“30”为根的树嵌入向量 $\mathbf{t}_l$ ，并基于 $\mathbf{t}_l$ 生成右节点的目标向量 $\mathbf{q}_r$ ，同时将 $\mathbf{q}_r$ 压入栈。接着栈会弹出一个节点的目标向量，树形解码器基于该目标向量预测该节点为数字“18”，停止继续分裂，生成以右节点“18”为根的树嵌入向量 $\mathbf{t}_r$ 。最后，树形解码器基于 $\mathbf{t}_l$ 和 $\mathbf{t}_r$ 生成以节点“+”为根的树

嵌入 $\mathbf{t}$ 。至此，以“+”为根节点的左子树构建完毕，其右子树构建过程类似，不再赘述。

2.2.3 监督模块

现有模型通常只提取问题文本中的浅层语义信息，难以有效地区分描述相似但解表达式不同的MWP，导致生成相似的问题表示。为解决这一问题，本文引入了一个基于多头注意力机制的监督模块^[3]，分别判断题目与正确表达式、可能的错误表达式之间的数学一致性概率，通过监督损失朝最大化一致性概率的方向优化编码器，纠正混淆的编码表示，确保编码结果的可分离性。

在实现过程中，本文采用0.15的概率替换 Q 的正确表达式 A 中的数学符号，以生成不正确表达式 A_{negative} ，其用来表示另一个描述与 Q 类似但解表达式不同的问题。

首先，本文对正确表达式 $A=\{q_{a_1}, q_{a_2}, \dots, q_{a_m}\}$ 进行词嵌入，得到词嵌入表示 $E_A=$

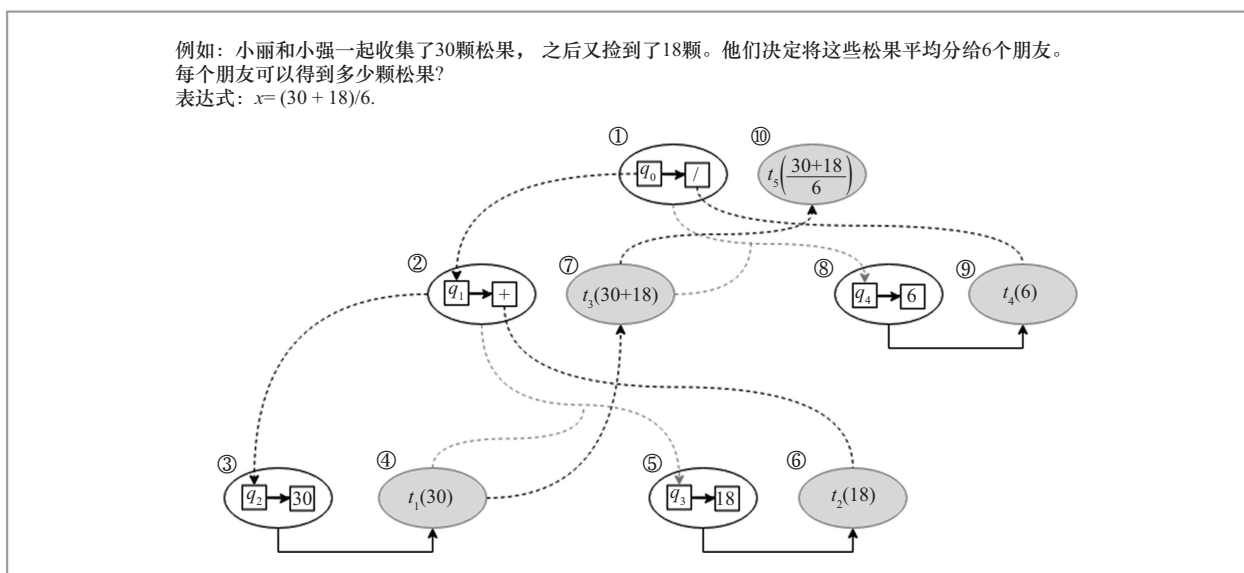


图2 UET构建过程举例

$\{e_{a_1}, e_{a_2}, \dots, e_{a_m}\}$ 。本文考虑到多头注意力机制通过将隐状态向量分成多个头, 形成多个子语义空间, 从而能够关注不同维度语义空间的信息。因此, 本文监督模块利用这一优势, 通过多头注意力机制丰富其语义表示, 获得表达式 A 的最终表示 H_A , 其中 $H_A = \{h_{a_1}, h_{a_2}, \dots, h_{a_m}\}$, h_{a_m} 为第 a_m 个元素的表示向量, 然后对 H_A 进行平均池化操作以获得整体语义表示向量 $\vec{h}^A$ 。同理可得到错误表达式 A_{negative} 的整体语义表示向量 $\vec{h}^{A_{\text{negative}}}$ 。

$$E_A = O_A W_{\text{Embed}} \quad (7)$$

$$H_A = \text{MultiHead}(E_A, E_A, E_A) \quad (8)$$

$$\vec{h}^A = \frac{1}{m} \sum_{i=1}^m h_{a_i} \quad (9)$$

其中, O_A 是独热向量矩阵, W_{Embed} 是可学习词嵌入矩阵。

然后, 监督模块将 MWP 编码表示 $Z = \{z_1, z_1, \dots, z_n\}$ 输入双向长短期记忆网络 Bi-LSTM, 以获得 Q 的高阶特征向量 $\vec{h}^Z$:

$$h_z = \text{BiLSTM}(Z) \quad (10)$$

$$\vec{h}^Z = \frac{1}{n} \sum_{i=1}^n h_{z_i} \quad (11)$$

最后, $\vec{h}^Z$ 分别与 $\vec{h}^A$ 、 $\vec{h}^{A_{\text{negative}}}$ 拼接并送入一致性检查模型, 输出问题与解表达式之间的数学一致性概率。

$$\text{Match}_{\text{predict}} = \text{ReLU}\left(\left[\vec{h}^Z, \vec{h}^A\right] W_M\right) V_m \quad (12)$$

其中, W_M 、 V_m 是可训练参数矩阵, ReLU 为非线性激活函数。

2.2.4 对比学习模块

本文使用对比学习方法, 旨在帮助 PLM 学习相同类型和不同类型 MWP 之间的异同, 提升模型对 MWP 之间相似性和差异性的理解。

首先, BSCL 为每个问题 Q 挑选一个具有相同类型和原型方程的正样本 Q_{positive} 。若问题 Q 在训练过程中找不到正样本, 则将自身视为正例。BSCL 再选择一道与 Q 不同类型的问题作为负样本 Q_{negative} 。

然后, BSCL 根据式 (1) 至式 (6), 获得整体语义表示向量 $\vec{Z}$ 、 $\vec{Z}_{\text{positive}}$ 、 $\vec{Z}_{\text{negative}}$; 再分别计算 $\vec{Z}$ 与 $\vec{Z}_{\text{positive}}$ 、 $\vec{Z}_{\text{negative}}$ 之间的相似度得分。

$$\text{sim}(\vec{Z}, \vec{Z}_{\text{positive}}) = \frac{\vec{Z} \cdot \vec{Z}_{\text{positive}}}{\|\vec{Z}\| \|\vec{Z}_{\text{positive}}\|} \quad (13)$$

其中, $\text{sim}(\vec{Z}, \vec{Z}_{\text{positive}})$ 表示向量 $\vec{Z}$ 与向量 $\vec{Z}_{\text{positive}}$ 的相似度得分, “ $\cdot$ ” 表示两个向量的点积, $\|\cdot\|$ 表示求向量的模。

2.2.5 优化目标

监督损失函数如下。

$$L_{\text{supervision}} = -\log P(\text{Match}_{\text{predict}} = 1 | Z, A, \theta) - \log P(\text{Match}_{\text{predict}} = 0 | Z, A_{\text{negative}}, \theta) \quad (14)$$

其中, θ 为监督模块中的模型参数。

对比学习损失函数如下。

$$L_{\text{CL}} = \max\left(0, \eta + \text{sim}(\vec{Z}, \vec{Z}_{\text{negative}}) - \text{sim}(\vec{Z}, \vec{Z}_{\text{positive}})\right) \quad (15)$$

其中, η 是超参数。

编码-解码损失函数如下。

$$L_{\text{Encoder-Decoder}}(\phi) = -\log P(A|Q) \quad (16)$$

其中, ϕ 是编码器和解码器中的模型参数。

BSCL 的总体损失函数如下。

$$L_{\text{total}} = L_{\text{Encoder-Decoder}} + \alpha \times L_{\text{CL}} - \beta \times \log P(\text{Match}_{\text{predict}} = 1 | Z, A, \theta) \quad (17)$$

其中, α 表示对比学习损失的权重, β 表示监督模块损失的权重, 本文将 β 设为 0.05。

3 实验与分析

3.1 数据集、基线方法与评估指标

为了验证 BSCL 在不同语言背景 and 不同类型 MWP 上的通用性和有效性, 本文在中文数据集 HMWP 和英文数据集 ALG514^[24] 上进行了大量实验。其中, HMWP 共包含 3 856 道 AWP 以及 1 614 道 ESP, ALG514 共包含 190 道 AWP 以及 324 道 ESP。

本文选择以下模型作为对比模型。

- GTS^[3]: 采用 Bi-GRU 作为编码器, 并通过树结构的解码方式生成通用 UET。

- Stack-Decoder^[7]: 使用 Bi-LSTM 作为编码器, 并采用基于栈的解码方法生成解表达式。

- GroupATT^[25]: 采用 Seq2Seq 结构, 在编码端提取应用题相关特征来丰富编码表示。

- DNS^[26]: 模型首先通过识别机制判断题目中的数字是否出现在解表达式中, 以去除冗余条件并保留有用数字, 随后使用 Seq2Seq 结构生成包含有效数字和运算符的数学表达式。

- SAU-Solver: 提出通用 UET 和一种具有语义对齐正则化的通用树解码方法。

- RoBERTa2UET: 使用预训练语言模型 RoBERTa 作为编码器, 使用树形解

码方式生成通用 UET。

- ChatGLM-6B^[27]: 基于 GLM (general language model) 架构的开源对话语言模型, 具有 62 亿个参数。本文使用 P-tuning v2^[28] 方法微调模型。

- ChatGLM2-6B: 是 ChatGLM-6B 的第二代版本。

本文选择使用等式准确率和答案准确率作为评估指标。

(1) 等式准确率

$$\text{Acc}_{\text{eq}} = \frac{\sum_{i=1}^{|T|} \text{pre}_{\text{eq}_i} = \text{GT}_{\text{eq}_i}}{|T|} \quad (18)$$

(2) 答案准确率

$$\text{Acc}_{\text{ans}} = \frac{\sum_{i=1}^{|T|} \text{pre}_{\text{ans}_i} = \text{GT}_{\text{ans}_i}}{|T|} \quad (19)$$

其中, pre_{eq_i} 表示模型预测的第 i 个问题的解表达式, GT_{eq_i} 表示实际解表达式, $\text{pre}_{\text{ans}_i}$ 表示根据 pre_{eq_i} 计算的数值答案, GT_{ans_i} 表示真实数值答案, $|T|$ 表示样本数量。

3.2 实验设置

本文使用 PyTorch 实现 BSCL, 并在 4 台 NVIDIA TESLA A800 上进行实验。实验设置批大小为 32, 并使用 Adam^[27] 优化器确保训练的稳定性。除编码器的初始学习率设置为 3×10^{-5} 外, 其他模块的初始学习率设置为 1×10^{-3} 。Dropout 设置为 0.5, 权重衰减设置为 1×10^{-5} 。监督模块的多头注意力个数设置为 8。在测试阶段, 使用大小为 5 的束搜索生成数学表达式。

3.3 实验结果与分析

各模型在 HMWP 和 ALG514 上的性能见表 2。

表2 各模型对比结果

模型	HMWP		ALG514	
	ACC _{eq}	ACC _{ans}	ACC _{eq}	ACC _{ans}
Stack-Decoder	—	0.274	—	0.289
GroupATT	0.252	0.332	0.331	0.376
DNS	0.240	0.327	0.370	0.418
GTS	0.316	0.420	0.204	0.547
SAU-Solver	0.323	0.429	0.208	0.549
RoBERTa2UET	0.370	0.500	0.212	0.619
BSCL	0.375	0.512	0.228	0.640

可以发现，本文提出的模型 BSCL 在两个数据集上的 ACC_{ans} 指标均优于其他基线模型，仅在 ALG514 数据集的 ACC_{eq} 指标上低于 GroupATT 和 DNS 模型。最后 4 个基于 Seq2Tree 的模型使用树结构解码构建通用 UET，在两个数据集上的 ACC_{ans} 表现均显著优于前 3 个模型。这突显了通用 UET 结构在建模不同类型的 MWP 方面的优越性。此外，本文采用 PLM 作为编码器的 RoBERTa2UET，在两个数据集上的 ACC_{ans} 分别比其他编码器的最佳模型 SAU Solver 高 0.071 和 0.070，进一步证明了 PLM 在理解不同类型 MWP 方面的出色表现。BSCL 和 RoBERTa2UET 之间的

比较充分验证了本文提出的对比学习方法和监督模块在求解 MWP 问题中的有效性，大幅提升了模型对 MWP 相似性和差异性的理解能力。

为了探索对话模型在求解不同类型应用题时的表现，本文在 HMWP 上对对话模型进行了实验。以下是使用 P-tuning v2 微调对话模型的具体方法。

- P2E&A：输入为问题的文本描述，输出为解表达式和数值结果。
- E2A：输入为问题的真实解表达式，输出为相应的数值结果。
- P2E-E2A：输入为问题的文本描述，输出为相应的解表达式。然后用训练好的 E2A 模型计算数值结果。
- P2E：输入为问题的文本描述，输出相应的解表达式，最后通过 Python 的 Sympy 科学计算库计算最终的数值结果。

由表3可知，ChatGLM2-6B 的答案准确率达 0.483，但仍比 BSCL 低 0.029，这进一步突显了 BSCL 在处理应用题方面的优越性。

表4展示了 BSCL 的消融实验结果，旨在验证本文提出模块的有效性。在去除对比学习或监督模块时，模型的求解精度有所下降，其表明这些组件在学习问题表示

表3 对话模型求解结果

模型	ChatGLM-6B		ChatGLM2-6B	
	ACC _{eq}	ACC _{ans}	ACC _{eq}	ACC _{ans}
P2E&A	0.300	0.257	0.366	0.326
E2A	—	0.483	—	0.487
P2E-E2A	0.341	0.368	0.370	0.404
P2E	0.341	0.441	0.370	0.483
BSCL	0.375	0.512	0.375	0.512

和识别各种类型应用题方面发挥了重要作用。

本文在 ALG514 数据集上进行了对比学习的超参数寻优实验,实验结果见表5。

由表5可知,当超参数 $\eta=0.2$ 和对比学习损失权重 $\alpha=0.5$ 时,BSCL 在 ALG514 上表现最佳。

本文还在 HMWP 和 ALG514 数据集上对 SAU Solver、BSCL 和 RoBERTa2UET 进行了5次交叉验证实验,并汇总了测试结果,实验结果见表6。

如表6所示,在各种实验中,使用 PLM 作为编码器的 RoBERTa2UET 和 BSCL 的表现均优于使用 Bi-GRU 作为编码器的 SAU-Solver。这突显了 PLM 在理解不同类型应用题方面的卓越能力。此外,通过对比 RoBERTa2UET 和 BSCL,进一步证明了本文提出的对比学习方法和监督模块在解决 AWP 和 ESP 时对模型性能的积极影响。

4 总结与展望

本文提出了一个可以有效解决不同类型 MWP 的模型 BSCL。本文通过引入验证数学一致性的监督模块和选择不同类型 MWP 作为负样本的对比学习模块,进一步增强了模型对 MWP 的深层语义以及隐含数学逻辑的理解,有效增强了其在处理

表4 BSCL消融实验结果

模型	HMWP		ALG514	
	ACC _{eq}	ACC _{ans}	ACC _{eq}	ACC _{ans}
BSCL w/o CL	0.370	0.506	0.227	0.620
BSCL w/o Supervision	0.371	0.508	0.215	0.634
BSCL	0.375	0.512	0.228	0.640

表5 对比学习参数实验结果

η	ACC _{eq}	ACC _{ans}	α	ACC _{eq}	ACC _{ans}
$\eta=0.1$	0.223	0.639	$\alpha=0.1$	0.218	0.628
$\eta=0.2$	0.228	0.640	$\alpha=0.5$	0.228	0.640
$\eta=0.3$	0.208	0.638	$\alpha=1$	0.225	0.631

不同 MWP 时的表现性能。

在未来的实际应用中,可以将本文提出的模型融入学习平台。学生通过平台的前端界面输入应用题后,系统会立即通过求解模型进行处理,并及时将解表达式和数值答案返回给学生,帮助他们理解题目的数学逻辑和解题思路。教师则可以通过查看学生平台上的答题记录,更好地了解学生的学习状况,进一步提升教学效果。这种学习平台不仅可以减轻教师的教学负担,还可以实现个性化辅助教学。然而,本文所提模型目前在可解释方面存在不足,在未来研究计划中,笔者将侧重提升模型的可解释性,考虑引入解释性机制,以揭示模型在求解应用题时依赖的关键信

表6 各模型在不同类型的问题中的答案准确率

模型	HMWP		ALG514	
	AWP-ACC _{ans}	ESP-ACC _{ans}	AWP-ACC _{ans}	ESP-ACC _{ans}
SAU-Solver	0.449	0.379	0.400	0.636
RoBERTa2UET	0.514	0.482	0.447	0.719
BSCL	0.521	0.491	0.474	0.738

息和决策标准。此外,笔者还计划将领域知识与自然语言理解及推理相结合,进一步提高模型求解跨学科应用题的能力。

参考文献:

- [1] ZHANG D X, WANG L, ZHANG L M, et al. The gap of semantic parsing: a survey on automatic math word problem solvers[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(9): 2287-2305.
- [2] QIN J H, LIN L H, LIANG X D, et al. Semantically-aligned universal tree-structured solver for math word problems[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg: USAACL, 2020: 3780-3789.
- [3] XIE Z P, SUN S C. A goal-driven tree-structured neural model for math word problems[C]//Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. Macao: IJCAI, 2019: 5299-5305.
- [4] ZHANG Y J, ZHANG T C, XIE P, et al. BEM-SM: a BERT-encoder model with symmetry supervision module for solving math word problem[J]. Symmetry, 2023, 15(4): 916.
- [5] WANG Y, LIU X J, SHI S M. Deep neural solver for math word problems[C]//The 2017 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: ACL, 2017: 845-854.
- [6] WANG L, ZHANG D X, ZHANG J P, et al. Template-based math word problem solvers with recursive neural networks[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Hawaii: AAAI Press, 2019, 33(1): 7144-7151.
- [7] CHIANG T R, CHEN Y N. Semantically-aligned equation generation for solving and reasoning math word problems[C]//Proceedings of the 2019 Conference of the North. Stroudsburg: ACL, 2019: 2656-2668.
- [8] LIANG Z W, ZHANG X L. Solving math word problems with teacher supervision[C]//Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence. Montreal: IJCAI, 2021: 3522-3528.
- [9] ZHANG J P, WANG L, LEE R K, et al. Graph-to-tree learning for solving math word problems[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2020: 3928-3937.
- [10] ZHANG J P, LEE R K, LIM E P, et al. Teacher-student networks with multiple decoders for solving math word problem[C]//Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence. New York: ACM, 2021: 4011-4017.
- [11] WU Q Z, ZHANG Q, FU J L, et al. A knowledge-aware sequence-to-tree network for math word problem solving[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg: [s.n], 2020: 7137-7146.
- [12] LIN X, HUANG Z Y, ZHAO H K, et al. HMS: a hierarchical solver with dependency-enhanced understanding for math word problem[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Virtually: AAAI Press, 2021, 35(5): 4232-4240.

- [13] LAN Y H, WANG L, ZHANG Q Y, et al. MWPToolkit: an open-source framework for deep learning-based math word problem solvers[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Virtually: AAAI Press, 2022, 36(11): 13188–13190.
- [14] ZHOU Z H, NING M Z, WANG Q F, et al. Learning by analogy: diverse questions generation in math word problem[C]//Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023. Stroudsburg: ACL, 2023: 11091–11104.
- [15] LIANG Z W, ZHANG J P, GUO K H, et al. Compositional mathematical encoding for math word problems[C]//Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023. Stroudsburg: ACL, 2023: 10008–10017.
- [16] LIANG Z W, ZHANG J P, WANG L, et al. Generalizing math word problem solvers via solution diversification[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Washington: AAAI Press, 2023, 37(11): 13183–13191.
- [17] LI Y Y, WANG L, KIM J J, et al. On the selection of positive and negative samples for contrastive math word problem neural solver[C]//Proceedings of the 17th International Conference on Educational Data Mining. Atlanta: International Educational Data Mining Society, 2024: 96–106.
- [18] HADSELL R, CHOPRA S, LECUN Y. Dimensionality reduction by learning an invariant mapping[C]//Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2006: 1735–1742.
- [19] YANG Z H, CHENG Y, LIU Y, et al. Reducing word omission errors in neural machine translation: a contrastive learning approach[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2019: 6191–6196.
- [20] CLARK K, LUONG M T, LE Q V, et al. ELECTRA: pre-training text encoders as discriminators rather than generators[EB]. arXiv preprint, 2020, arXiv:2003.10555.
- [21] GAO T Y, YAO X C, CHEN D Q. SimCSE: simple contrastive learning of sentence embeddings[C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: ACL, 2021: 6894–6910.
- [22] YU Y, ZUO S M, JIANG H M, et al. Fine-tuning pre-trained language model with weak supervision: a contrastive-regularized self-training approach[C]//Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: ACL, 2021: 1063–1077.
- [23] LIU Y H, OTT M, GOYAL N, et al. RoBERTa: a robustly optimized BERT pre-training approach[EB]. arXiv preprint, 2019, arXiv:1907.11692.
- [24] KUSHMAN N, ARTZI Y, ZETTLEMAYER L, et al. Learning to automatically solve algebra word problems[C]//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2014: 271–281.
- [25] LI J R, WANG L, ZHANG J P, et al. Modeling intra-relation in math word problems with different functional

multi-head attentions[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2019: 6162–6167.

[26] WANG Y, LIU X J, SHI S M. Deep neural solver for math word problems[C]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: ACL, 2017: 845 – 854.

[27] DU Z X, QIAN Y J, LIU X, et al. GLM: general language model pretraining with autoregressive blank infilling[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Stroudsburg: ACL, 2022: 320–335.

[28] LIU X, JI K X, FU Y C, et al. P-tuning v2: prompt tuning can be comparable to fine-tuning universally across scales and tasks[EB]. arXiv preprint, 2021, arXiv: 2110.07602.

作者简介



张天成（1969–），男，博士，东北大学计算机科学与工程学院教授，主要研究方向为人工智能、时空数据管理和智慧教育。



王玉杨（2000–），女，东北大学计算机科学与工程学院硕士生，主要研究方向为辍学率预测、数学应用题的自动求解。



张亦嘉（1999–），男，东北大学计算机科学与工程学院硕士生，主要研究方向为数学应用题的自动求解、大型语言模型微调。



于明鹤（1989–），女，博士，东北大学软件学院副教授，主要研究方向为数据库、信息检索和智慧教育。



冷芳玲（1978- ），女，博士，东北大学计算机科学与工程学院讲师，主要研究方向为知识图谱、智慧教育。



于戈（1963- ），男，博士，东北大学计算机科学与工程学院教授，主要研究方向为数据库、区块链和智慧教育。

收稿日期: 2024-11-08
通信作者: 张亦嘉, neu_yjzhang@163.com
基金项目: 国家自然科学基金项目 (No.62272093; No.62372097; No.62137001)
Foundation Items: The National Natural Science Foundation of China (No.62272093, No.62372097, No.62137001)