Hindawi 计算智能与神经科学 2022 年卷，
文章编号 7839840，共 21 页
https://doi.org/10.1155/2022/7839840

研究论文

RLAS-BIABC：采用经改进人工蜂群算法增强的 BERT 模型进行基于强化学习的答案选择

Hamid Gharagozlou，[1] JavadMohammadzadeh，[1] Azam Bastanfard，[1]和 Saeed hiry Ghidary [2]

[1]伊斯兰阿扎德大学卡拉季分校计算机工程系，伊朗卡拉季[2]斯塔福德郡大学数字、技术与艺术学院，英国特伦特河畔斯托克

通信请联系 Javad Mohammadzadeh；j.mohammadzadeh@kiau.ac.ir

收稿日期：2022年1月23日；修订日期：2022年3月23日；接受日期：2022年4月4日；发布日期：2022年5月6日

学术编辑：Ripon Chakrabortty

版权 © 2022 Hamid Gharagozlou 等。本文为开放获取文章，依据知识共享署名许可协议发布。该协议允许在任何媒介中不受限制地使用、分发和复制，但须适当注明原始作品。

答案选择（AS）是开放域问答（QA）问题中的一项关键子任务。本文提出了一种名为 RLAS-BIABC 的答案选择方法，该方法基于注意力机制和长短期记忆网络（LSTM），并采用基于变换器的双向编码器表示（BERT）词嵌入；此外，还引入改进的人工蜂群算法进行预训练，并采用基于强化学习的算法训练反向传播（BP）算法。BERT 可融入下游任务并进行微调，形成统一的特定任务架构；预训练的 BERT 模型能够捕捉不同的语言特征。现有算法通常使用正负样本对训练答案选择模型，以构建二分类器。正样本对包含一个问题和一个真实答案，而负样本对则包含一个问题和一个虚假答案。正样本对的输出应为 1，负样本对的输出应为 0。通常负样本对多于正样本对，导致分类不平衡，进而大幅降低系统性能。为解决这一问题，我们将分类定义为一个序贯决策过程：智能体在每个步骤获取一个样本并对其进行分类。每次分类操作后，智能体都会获得奖励，其中多数类的奖励低于少数类的奖励。最终，智能体会找到策略权重的最优值。我们使用改进的人工蜂群算法初始化策略权重。这种初始值设定技术可以避免陷入局部最优等问题。尽管人工蜂群算法在大多数任务中表现良好，但它仍存在一个弱点：在发现邻近食物源位置时，忽略了相关个体对的适应度。因此，本文还提出了一种互学习技术，根据互学习因子，在两个选定个体中选取适应度较高者，以修改生成的候选食物源。我们在 LegalQA、TrecQA 和 WikiQA 三个数据集上测试了模型，结果表明，RLAS-BIABC 可被视为一种最先进的方法。

1. 引言

如今，网络空间中的 Stack Overflow 和 GitHub 等众多领域每天都会涌现大量问题。问答（QA）是自然语言处理（NLP）的重要分支之一，能够自动回答问题。问答可以通过两种方式实现：一些方法侧重于生成答案，通常会采用生成对抗网络（GAN）等网络来生成答案 [1]。然而，这些方法无法保证答案在语义和语法上的准确性。

另一类方法采用答案选择，这是问答的一项重要子任务，也应用于机器阅读理解等其他领域 [2]。过去几年，这一问题日益受到关注[3, 4]。给定一个问题 q 和一组候选答案 $A = \{a_1, a_2, a_3, \ldots, a_N\}$，目标是找出 $a_i \in A$作为问题 q的最佳答案。问题和答案的长度各不相同，而且一个问题可能有多个正确答案。

从文献来看，存在许多基于传统方法和深度学习方法的 AS方法 [5]。

传统方法更多地依赖搜索引擎 [6], 信息检索 [7, 8], 、手工规则 [9], 或机器学习模型 [10, 11]。基于信息检索的模型基于关键词进行工作，不使用任何语义数据，因此难以获得正确的答案 [12]。基于手工规则的技术无法涵盖所有模式，其性能受到限制 [13, 14]。在基于机器学习的方法中，特征由人工构建，因此其质量很大程度上取决于特征提取 [15, 16]。一些标准和分类器（包括编辑距离和支持向量机）会考虑AS对之间的匹配关联 [11]。通常，传统方法存在两个主要缺陷。首先，它们大多不使用关键词、特征或规则中的语义信息，因而无法考虑问答对之间的全方位关系。其次，特征提取和手工规则缺乏灵活性，导致泛化能力较差。深度学习出现后，许多领域中的诸多问题[17–23], （包括AS）都因其发展而得到改善。用于AS的基于深度学习的方法通常采用卷积神经网络（CNN） [24]或LSTM，以提取不同层次的语义特征。其主要任务是估计问答对之间的语义相似度，这可以视为文本相似度计算或分类工作。CNN用于建模句子的层次结构，并评估其匹配程度[25]。与此同时，LSTM用于生成问题和答案的嵌入，同时保留序列依赖信息。尽管深度模型只能实现有限的改进，但它们仍面临一些困难。它们通过一种神经网络设计构建问答对的嵌入表示。这导致模型关注单方面的特征，而忽略问答对之间其他复杂的语义特征。随后，试图理解语言的模型得到发展 [26]。这些模型通过不同的方法学习语言的句法和语义规则，包括下一个词和句子预测以及掩码词预测 [27]。它们能够识别语言，并根据正确的句法和语义规则生成新文本。BERT模型 [27] 是最新的语言模型之一，性能优于其他已开发的语言模型。该模型利用了Transformer中提出的表述 [28], ，而Transformer目前已广泛应用于NLP任务 [29]。

深度模型的成功主要取决于架构、训练算法以及训练中所用特征的选择。这些因素使深度网络的设计成为一个复杂的优化问题 [30]。在许多方法中，拓扑结构和传递函数是固定的，所有可能的权重和值偏置构成了潜在网络的搜索空间 [31]。在 [32, 33] 中， [34], 蚁群优化 [35], 禁忌搜索 [36], 模拟退火[37], 和遗传算法 [38] 被用于训练拓扑结构固定的神经网络。神经网络学习的优化过程旨在找到能够使输出误差最低的权重配置。然而，深度模型最优权重的确定在很大程度上取决于权重初始化，其对神经网络性能的影响比网络架构和训练样例更为显著 [39]。答案选择方法，包括深度方法，采用基于梯度的算法（如反向传播算法（BP）和列文伯格–马夸尔特（LM））进行模型权重优化 [40]。BP算法通过一阶导数收敛，而LM算法则通过二阶导数收敛 [41]。BP和LM的主要问题在于对初始权重敏感，这会导致陷入局部优化 [42]。为了解决这一问题，人们采用具有逃离局部极小值能力的全局搜索方法来预训练权重，例如基于群体的元启发式（PBMH）算法[43–45]。在PBMH算法中，人工蜂群算法是用于优化问题的最强大算法之一，与传统算法相比具有两大优势：无需计算梯度，也不会陷入局部优化 [46]。该算法基于蜜蜂的智能行为，包含两个基本概念：食物源和人工蜂。人工蜂会寻找花蜜丰富的食物源。食物源的位置代表优化问题的一个解，而花蜜的数量则等同于解的质量。尽管食物源位置是决定蜜蜂是否选择某一食物源的关键因素，但蜜蜂生成邻近食物源时仍缺少一些必要信息。

答案选择中的另一个主要问题是不平衡类别：正类（包括问题及其对应答案）的样本数量远少于负类（包括问题及其不对应的答案），这会降低现有方法的性能。针对不平衡问题提出的方法通常分为两类：数据级方法和算法级方法。在数据级算法中，通过对少数类进行过采样、对多数类进行欠采样，或同时采用这两种方法，调整训练数据以平衡类别分布。SMOTE [47]是一种过采样方法，通过对相邻的少数类样本进行线性插值来生成新样本。Near Miss[48] 是一种欠采样方法，通过随机删除较大类别中的样本来处理不平衡问题。当两个不同类别的数据点在分布上彼此接近时，该算法会删除较大类别中的数据。过采样算法可能增加过拟合的风险，而欠采样算法则会丢失多数类中的有价值的信息。在算法级方法中，采用代价敏感学习、集成学习和决策阈值调整等技术来提高少数类的重要性。在代价敏感学习方法中，损失函数会为每个类别的错误分类分配不同的代价，其中少数类的代价更高。基于集成学习的解决方案会训练多个子分类器，并采用投票来获得更好的结果。阈值调整技术会在不平衡数据集上训练分类器，并在测试期间更改决策阈值。人们还提出了基于深度学习的方法来分类不平衡数据。论文 [49] 提出了一种损失

函数，使深度模型对多数类和少数类的分类错误给予同等权重。另一项研究 [50] 在保持簇间和类间间隔的同时，学习不平衡数据的判别特征。 [51] 中的作者提出了一种基于自助采样算法的方法，可在每个小批量中平衡卷积网络的训练数据。 [52] 提出了一种用于优化网络权重和类别敏感代价的算法。在 [53]，中，作者提取了少数类中的复杂样本，并通过采用类别校正损失函数 [54] 进行批次优化，改进了他们的算法。

近年来，深度强化学习已成功应用于电脑游戏、机器人控制、推荐系统 [55–57]，等领域。对于分类问题，深度强化学习有助于消除噪声数据并学习更优的特征，从而显著提升分类性能。然而，针对将深度强化学习应用于不平衡分类的研究仍然很少。深度强化学习非常适用于不平衡分类，因为其学习机制和特定奖励函数能够通过给予少数类更高的奖励或惩罚项，使模型更多地关注少数类。

本文提出了一种基于注意力机制的 AS 长短期记忆模型，称为 RLAS-BIABC。该模型基于 BERT 词嵌入、强化学习和改进的人工蜂群算法构建。RLAS-BIABC 模型的主体由两个基于注意力机制的双向长短期记忆网络（BLSTM）和一个用于计算问答对相似度的前馈网络组成。该模型旨在学习正样本对和负样本对。正样本对由问题和真实答案构成，而负样本对则将每个问题与其他答案配对。我们使用 BERT 作为词嵌入，在不使用预先设计的特征的情况下学习句子之间的语义相似度。此外，我们为 RLAS-BIABC 引入了一种改进的人工蜂群算法，用于为所有长短期记忆网络、注意力机制和前馈网络寻找权重初始化值，以启动反向传播算法。为此，我们通过在两个选定的位置参数之间进行互学习来改进人工蜂群算法，从而选择适应度更高的候选食物源。此外，在反向传播步骤中，我们提出的方法采用强化学习来处理不平衡分类问题。为此，我们将 AS 问题定义为一个猜谜游戏，并将其划分为一个序贯决策过程。在每一步中，智能体接收由一个训练实例表示的环境状态，然后在策略的指导下执行二分类操作。如果分类器成功完成操作，就会获得正奖励；否则，就会获得负奖励。少数类获得的奖励高于多数类。智能体的目标是在序贯决策过程中尽可能多地获得累积奖励，也就是说，尽可能准确地对样本进行分类。我们在三个标准数据集 TrecQA、LegalQA 和 WikiQA 上评估 RLAS-BIABC 模型，结果表明，RLAS-BIABC 优于其他采用随机加权的方法。

本文的主要贡献如下：（1）我们采用了 BERT 词嵌入，这是目前最新的多语言模型。（2）我们没有使用随机权重系统来设置模型权重，而是定义了一种编码策略，并使用改进的人工蜂群算法计算初始值。（3）我们将 AS 问题视为一个序贯决策过程，并提出了一个用于不平衡分类的深度强化学习框架。（4）我们通过实验研究所提模型的性能，并将其与其他采用随机权重进行初始化、且面临不平衡分类问题的方法进行比较。

本文其余部分安排如下：第2节简要回顾相关研究。第3节介绍人工蜂群算法。第4节描述所提出模型的框架。第5节展示评估指标、数据集和结果。第6节总结全文并展望未来工作。

2. 相关工作

迄今为止，针对问答问题已提出许多方法。本节概述基于机器学习和深度学习的方法。

最早提出的方法基于特征工程。在这些方法中，通过重复出现的常用词来衡量问题与答案之间的关系，其中词袋和词组袋 [58] 常用于此目的。这些方法不符合逻辑，因为它们没有考虑句子中的语义和语言特征。随后，一些研究利用 WordNet [59] 等语言资源解决语义问题，但未能消除语言方面的局限。一些研究人员考虑了句子的句法和语义结构 [60]。一些作者采用了依存树和树编辑距离算法[15, 61]。研究 [62] 证实，WordNet 和命名实体识别 [63] 等工具在选择语义特征方面可以发挥重要作用。该文章 [64] 为自动特征选择提供了一种有效的解法。这些方法是消除特征工程的早期尝试之一。

后来，随着深度学习的兴起，许多方法将深度模型用作自动特征工程工具。近年来，深度学习已广泛应用于自然语言处理领域[18]。此外，循环神经网络 (RNN) 和卷积神经网络 (CNN) 作为深度学习的两大有力工具，被用于特征提取 [20, 21]。深度学习方法处理问答对的方式分为两类。第一类将问题和答案视为两个不同的元素，并通过深度网络分别获得它们的表示向量。通常采用各种度量标准来衡量二者之间的相似度。文献 [65] 的作者提出了一种比较聚合系统，采用多种度量标准来衡量相似度。研究 [66]利用 ELMo 语言模型 [26] 来处理问题和答案。结果表明，语言模型具有优越性。第二类则将问题和答案

视为一个单独的句子。文献 [67],提出了一种基于 CNN 的方法，以点式方式为问答对评分。文献 [68] 中的另一种方法将双向长短期记忆网络用于问答。首先学习问题词语和答案词语的嵌入，然后将其输入双向长短期记忆网络，接着根据每个句子中词语嵌入的平均值估算该句子的嵌入。最后，将答案与问题之间的关联输入前馈网络。孪生网络 [69] 是深度学习的重要分支，已应用于各个领域，尤其是问答领域。该网络为问题和答案分别提供两个表示向量。文献 [70], 首次提出了用于答案选择任务的深度学习任务。在这项研究中，使用卷积神经网络和逻辑回归提取与问题最相关的答案。研究 [71] 实现了文献 [70] 提出的构想。作者尝试利用隐藏层、卷积操作和激活函数构建不同的模型，以改善结果。文献 [72] 中的另一项工作结合多种模型，为每个句子生成表示向量。文献 [73], 将每个点式模型转换为成对模型。其设想是，成对模型能够进一步提升模型性能。文献 [72]也将成对模型应用于该模型。研究[74] 是一项预处理操作。在这项研究中，命名实体被替换为唯一的词元，从而便于选择候选答案。将该技术应用于文献 [73]提出的模型后，证实了其显著效果。同时，文献 [75]的作者指出，并非所有命名实体都能用同一个词元替换，因此他们为每个命名实体指定一个词元。后来发现，使用注意力机制可以构建更有效的模型。与基于孪生网络的方法不同，注意力机制利用问题与答案之间的上下文敏感交互 [76] 。注意力机制最初是为机器翻译提出的，后来被用于问答等其他应用 [77, 78]。文献 [79] 提出的方法结合注意力机制和循环神经网络，以完成答案选择任务。该方法基于文献 [80]提出的注意力机制。在文献 [81], 中，作者采用基于交互加权对齐网络的方法来确定问答对之间的相似度。文献 [82] 提出了一种基于双向对齐机制和堆叠循环神经网络的方案。在早期研究中，注意力机制仅应用于循环神经网络；但后来，文献 [83] 指出，将卷积神经网络与注意力机制结合可能更加高效。

3. 背景

*3.1.*长短期记忆网络（*LSTM*）。简而言之，循环神经网络[84] 旨在对序列输入进行建模。在这些网络中，数据序列被映射为一系列隐藏状态。随后，使用以下公式生成输出：

$$h_t = \theta\left(W_h h_{t-1} + U_h x_t + b_h\right). \tag{1}$$

$$y_t = \tau\left(W_y h_t + b_y\right), \tag{2}$$

其中， W_h 和 U_h 是权重矩阵， b 表示偏置。 θ和 τ 表示激活函数，例如 ReLU 和双曲正切函数。$x_t \in \mathrm{R}^d$ 是维度为 d 的输入，而 $h_t \in \mathrm{R}^h$表示时间 t时大小为 h 的隐藏层。

循环神经网络已在自然语言处理的许多领域取得成功，例如文本生成 [85] 和文本摘要[86]。然而，后来人们发现，随着这些网络的输入长度增加，它们会遭遇梯度爆炸和梯度消失 [87]等问题。Hochreiter 和 Schmidhuber 提出的长短期记忆网络 [88] 能够避免上述问题，这是因为记忆单元可以有效处理长程依赖。具体而言，长短期记忆网络由多个控制门和一个记忆单元组成。令 x_t、 h_t 和 c_t 分别表示时间 t的输入、隐藏状态和记忆细胞。给定一个输入序列（x_1， x_2，..... x_T），长短期记忆网络应计算并输出一个隐藏单元序列（h_1， h_2，..... h_T）和一个记忆细胞序列（c_1， c_2，..... c_T）。用公式表示，该过程可定义如下 [89]:

$$\begin{aligned}
i_t &= \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right),\\
f_t &= \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right),\\
c_t &= f_t c_{t-1} + i_t \tanh\left(W_j x_t + U_j h_{t-1} + b_j\right),\\
o_t &= \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right).\\
o_t &= \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right),\\
h_t &= o_t \tanh\left(c_t\right),
\end{aligned} \tag{3}$$

其中， W 和 b 为网络参数。 i、 f和 o 分别表示输入门、遗忘门和输出门。 σ代表 Sigmoid 函数。

尽管许多问题都可以在长短期记忆网络的框架下得到解决， [18, 19, 90], 实验表明，双向长短期记忆网络可能比长短期记忆网络更有效。双向长短期记忆网络 [91] 是一种扩展的长短期记忆网络，能够从头到尾以及反向处理输入。这一过程会针对 t时刻的特定输入生成两个隐藏向量，即 $\overrightarrow{h}_t$ 和 $\overleftarrow{h}_t$。因此，连接后的向量，即 $[\overrightarrow{h}_t, \overleftarrow{h}_t]$，构成最终的隐藏向量。

长短期记忆网络中的各单元提取的信息在做出最终决策时具有同等重要性，这会降低系统性能。为说明这一点，请看这个句子：“尽管她来自北方邦，但由于在孟加拉长大，她的孟加拉语很方便。”在这个句子中，“孟加拉语”和“孟加拉”等词应受到更多关注，而原始长短期记忆网络并非如此。为解决这一问题，我们引入了注意力机制。在注意力机制系统中，每个隐藏层的重要性由一个位于区间 [0, 1] 内的系数表示，并参与最终向量的构建。形式上，对于一个

长度为 T 的特定输入，考虑每个隐藏向量 h_t 对应的系数 α_t，其隐藏单元向量计算如下：

$$h = \sum_{t=1}^{T} \alpha_t h_t. \tag{4}$$

3.2. 人工蜂群（*ABC*）算法。人工蜂群算法是一种受自然界蜜蜂智能行为启发的技术。人工蜂群算法主要由两个基本概念构成：食物源和人工蜂。人工蜂寻找花蜜丰富的食物源。食物源的位置表示优化问题的一个解，而花蜜的数量则对应于解的质量。人工蜂群算法涉及三类不同的蜜蜂：雇佣蜂、观察蜂和侦察蜂。雇佣蜂在其周围其他食物源附近寻找花蜜更多的食物源，并在舞蹈区与观察蜂分享信息。雇佣蜂和观察蜂的数量相同，各占群体总数的一半。每只雇佣蜂都驻于一个蜂巢中，因此雇佣蜂的数量等于蜂巢总数。与雇佣蜂一样，观察蜂也会在邻近区域寻找最佳食物源。若雇佣蜂的食物源经过若干步后仍未改善，则会转变为侦察蜂，并开始新的搜索。人工蜂群算法的优化过程总结如下：

初始化阶段。搜索空间中的食物源（即蜜蜂的位置）按如下方式初始化：

$$x_i^j = x_{\min}^j + rand\,(0.1)\left(x_{\max}^j - x_{\min}^j\right). \tag{5}$$

其中，i 表示第 i个解，其取值为区间 $[1, BN]$内的整数，其中BN 为解的总数。每个解由 D 个元素组成，其中 D 表示待优化的权重数量。 x^j 最小值和 $x_{\max}^j$ 分别为解 i中的最小值和最大值。

雇佣蜂阶段。初始化后，雇佣蜂会在现有食物源的邻域中寻找新的食物源。随后，它们计算指定食物源的质量。如果新食物源的质量更好，就会从记忆中删除先前食物源的信息，并用新食物源的信息替换。否则，先前食物源的信息将保持不变。形式上，这一步可由以下公式描述：

$$v_i^j = x_i^j + \varphi_i^j\left(x_i^j - x_k^j\right), \tag{6}$$

其中，k 在区间 $[1, BN]$ 内取整数值，φ_i^j 是 $[-1, 1]$ 内的随机小数值，而 v_i 是通过改变元素 x_i 得到的新食物源。观察蜂阶段。在此阶段，雇佣蜂向观察蜂提供信息。观察蜂计算信息的值，并根据概率值选择新的解。与前一步一样，如果新解含有更多花蜜，则用新解替换之前的位置信息。选择新解的可能性可表示如下：

$$p_i = \frac{\mathrm{fit}(x_i)}{\sum_{n=1}^{BN} \mathrm{fit}(x_n)}, \tag{7}$$

其中，fit(xi)是第 i个解的适应度值。根据式（7），fit(xi)越高，观察蜂接受该解的可能性就越大。如果选择了该解，观察蜂便会前往该食物源，并根据式（6）生成一个新解。

侦察蜂阶段。最后一步中，使用侦察蜂来摆脱局部最优。更具体地说，经过若干轮迭代仍未能改善的解会转变为侦察蜂，并丢弃其食物源。因此，根据（6）用新的食物源替换旧的食物源。

重复执行上述四个步骤，直至满足终止准则。完整的人工蜂群算法见算法1。

4. RLAS-BIABC框架

所提出的算法针对分类考虑了两个关键选项。首先，我们构建一个包含模型中所有可学习权重的向量，并使用人工蜂群算法对其进行优化。随后，在其余流程中应用反向传播算法。此外，包括我们的方法在内的大多数分类器还面临另一个问题，即数据不平衡。为考虑这一因素，我们采用强化学习的思路。我们在两个独立的章节中介绍这两种方法。

所提出模型的整体架构如图1所示。设问题 Q 包含一个由n 个词组成的序列，其中 $Q = (q_1, q_2, \ldots q_n)$；答案为 A，其中 $A = (a_1, a_2, \ldots a_m)$，包含$m$ 个词。设a_i，$q_j \in \mathrm{R}^D$ 表示一个词的 D维视觉表示。问题和答案分别使用两个长短期记忆网络。使用两对正负样本来学习模型。在正样本对（Q，A）中，A 是问题 Q的正确答案；模型的输出应趋近于1。而在负样本对（Q，A'）中，A' 是该问题的虚假答案，网络对这一对的输出应趋近于0。长短期记忆网络计算得到的问题和答案嵌入表示如下：

$$\begin{aligned} q &= \sum_{i=1}^{n} \alpha_i h_{q_i}, \\ a &= \sum_{i=1}^{m} \beta_i h_{a_i}, \end{aligned} \tag{8}$$

其中，$h_{q_i} = [\overleftarrow{x}_i, \overrightarrow{x}_i]$ 和 $h_{a_i} = [\overleftarrow{y}_i, \overrightarrow{y}_i]$ 分别是与问题和答案相关的第 i 个双向长短期记忆网络的输出。 α_i和 β_i 是每个单元的注意力权重，计算如下：

输入：D：解的维度，BN：种群规模，limit：循环次数，MaxItr：最大迭代次数 (1) 使用 (5) 初始化解群体 X ← [$x_1, x_2, \ldots, x_{BN}$] (2) Itr ← 1. (3) **当** ≤MaxItr**时执行**(4) //雇佣蜂阶段 (5) **对**i ← 1 至 BN**执行**(6) 使用 (6) 生成新解 x_{new} (7) 计算 x_{new} 的适应度 f_{new} (8) 如果 x_i 更优，则用 x_i 替换 x_{new} (9) **结束循环**(10) 使用 (7) 计算X 中每个解的概率 p (11) //观察蜂阶段 (12) **对**i ← 1 至 BN**执行**(13) **如果** rand (0，1) < p_i**则**(14) 使用 (6) 生成新解 x_{new} (15) 计算 x_{new} 的适应度 f_{new} (16) 如果 x_i 更优，则用 x_i 替换 x_{new} (17) **结束条件判断**(18) **结束循环**(19) //侦察蜂阶段 (20) 如果发现被遗弃的解，则用通过 (6) 生成的解替换它 (21) 将迄今为止的最优解放入 x_{best} (22) Itr ← Itr + 1. (23) **结束循环**(24) **返回**x_{best}

Algorithm 1：人工蜂群算法的伪代码。

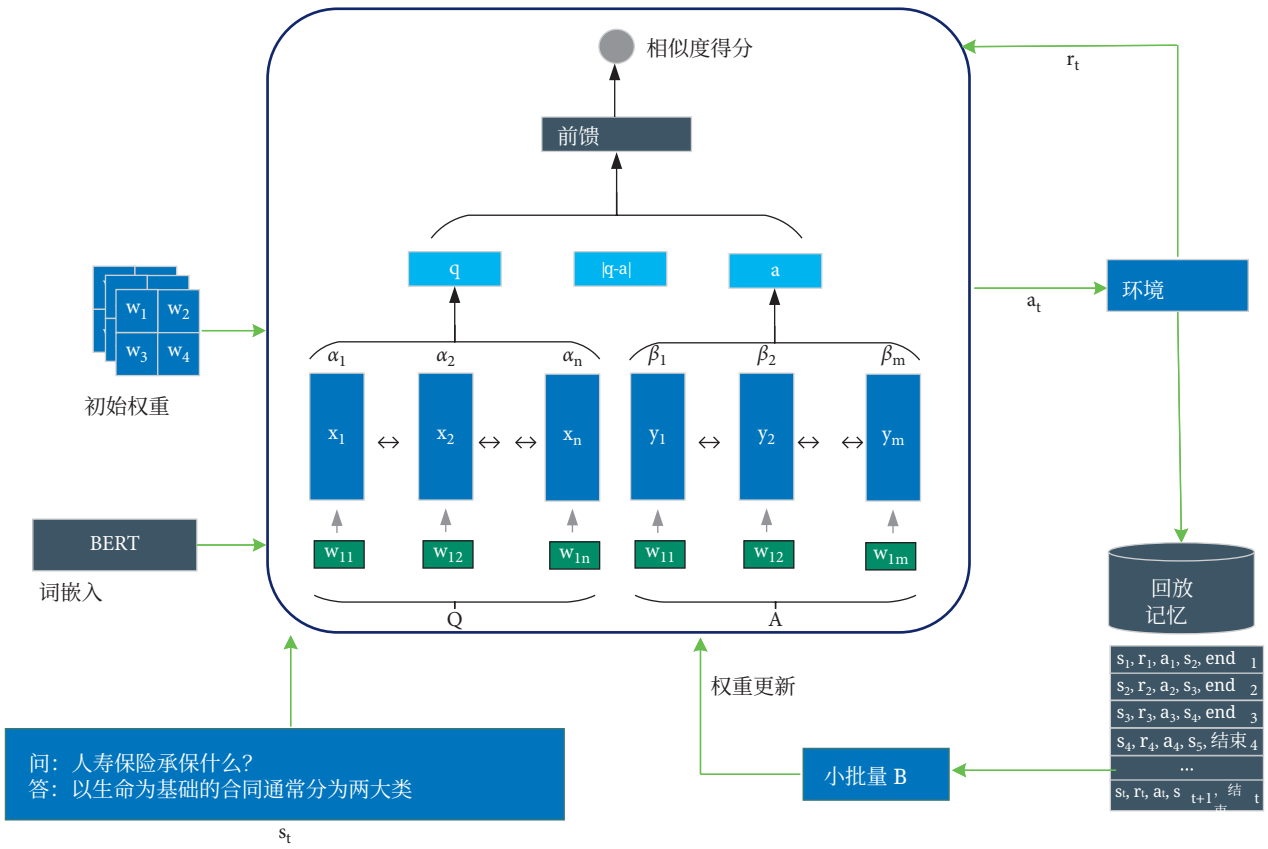


图1：所提出的 LSTM 相似度模型。

$$\alpha_i = \frac{e^{u_i}}{\sum_{i=1}^{n} e^{u_i}},$$
$$\beta_i = \frac{e^{v_i}}{\sum_{i=1}^{m} e^{v_i}},$$
$$u_i = \tanh\left(W_u\left[\overleftarrow{x}_i, \overrightarrow{x}_i\right] + b_u\right), \quad (9)$$
$$v_i = \tanh\left(W_v\left[\overleftarrow{y}_i, \overrightarrow{y}_i\right] + b_v\right),$$

其中，W_u，W_v，b_u，b_v表示注意力机制的参数。通过注意力机制确定问题和答案的有效表示后，我们按照图1将连接后的 q、a和 $|q-a|$ 组成一个向量，并将其输入前馈网络。实验已证实，两个表示向量之间的差异有助于做出成功的决策 [92]。

4.1. 基于 *BERT* 的词嵌入。词嵌入用于将词映射为语义向量，以供深度学习算法使用。词嵌入是一种可靠的方法，可以提取词在其上下文中形成的重要表示。许多研究致力于在 Skip-gram [93]、GloVe [94], 和 FastText [95]等神经网络模型中寻找最具意义的词表示。近几年来，预训练语言模型（PLM）得到了广泛应用。此类模型是一个包含自然语言先验知识的黑盒，并会在自然语言处理工作中进行微调。

PLM 模型通常使用未标注数据来学习模型参数 [96]。本文将 BERT 模型 [27],视为 PLM 发展趋势中的最新技术之一。BERT 是一种双向语言模型，使用维基百科等大型数据集进行训练，以生成上下文表示。此外，通常会针对不同的分类任务，对其神经网络全连接层进行微调。微调功能将上下文或特定问题的含义与预训练的通用含义相结合，并针对分类任务进行训练。

图 2 展示了 BERT 模型的架构。BERT 使用双向 Transformer，其各层中的表示会同时基于左右两侧的上下文 [97], , 这使其有别于 Word2Vec 和 GloVe 等其他模型；后者只沿一个方向构建嵌入，因此无法反映上下文差异。

4.2. 预训练阶段。权重初始化是设计神经网络时至关重要的一环；忽视这一点会导致模型产生误导性结果。所提出的结构包含两个长短期记忆网络、两个注意力机制和一个前馈神经网络，每个网络都具有需要训练的权重。本文采用改进的人工蜂群算法对权重进行预训练。

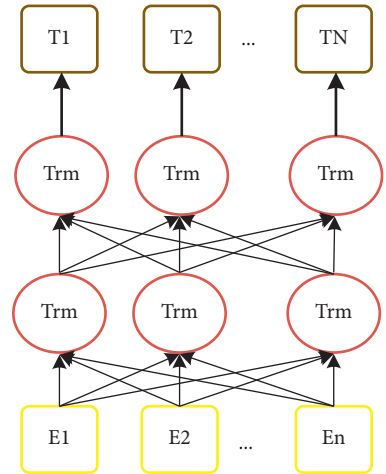


图 2：BERT 模型的架构。

4.2.1. 基于互学习的人工蜂群算法。工蜂会随机选择一个食物源位置，并对其进行更改以生成新位置。如果新解的适应度值更高，就会取代当前解；否则，不做任何更改。换言之，在一个 D 维优化问题中，每次迭代都会随机选择一个维度，更改其值，并选取较优的结果。根据式（6），新生成的解 v_i^j 仅取决于两个参数 x_i^j 和 x_k^j，因此食物源 v_i^j 无法得到控制，有时比当前食物源更大，有时则更小。在人工蜂群算法中，需要适应度值更高的食物源。为始终生成值更高的食物源，我们考虑通过当前食物源与相邻食物源之间的互学习获得的适应度信息。

$$v_i^j = \begin{cases} x_i^j + \varphi_i^j\left(x_k^j - x_i^j\right), & \mathrm{Fit}_i < \mathrm{Fit}_k \\ x_k^j + \varphi_i^j\left(x_i^j - x_k^j\right), & \mathrm{Fit}_i \geq \mathrm{Fit}_k \end{cases}, \quad (10)$$

其中，Fit_i 和 Fit_k 分别表示当前食物源和邻近食物源的适应度值。φ_i^j 表示区间 $[0, F]$内的均匀随机数，其中 F 是一个名为互学习因子的非负常数。由此可见，值v_i^j 取决于它们的位置及其适应度值。通过比较当前食物源和邻近食物源，新解的适应度值会朝着更优的食物源移动。也就是说，如果当前食物源的适应度更高，候选解就会朝着更优的解移动；否则，它将趋向于邻近食物源。这种学习策略有助于生成更优的候选解。参数 F在平衡相关食物位置之间的扰动方面发挥着重要作用。此外，F 必须是一个非负数，以确保解朝着更优的方向移动。随着 F从零增大到某个特定值，相应位置上的扰动会减小，这意味着新食物源的适应度值接近较高的适应度值。较大的 F 会削弱开发和探索的能力。

4.2.2. 编码策略。编码是指将权重排列成一个向量，并将其视为人工蜂群算法中蜜蜂的位置。选择合适的排列方式是一项具有挑战性的任务；不过，经过多次实验，我们尝试设计出尽可能理想的编码策略。图3展示了一个编码示例，其中包含两个长短期记忆网络、两个注意力机制和一个双层前馈网络。请注意，所有权重矩阵均按行存储。

4.2.3. 适应度函数。适应度函数旨在衡量一个解的效率。本文采用以下函数作为能力函数：

$$\text{fitness} = \frac{1}{1 + \sum_{i=0}^{T} (y_i - \tilde{y}_i)^2}, \tag{11}$$

其中，T 为训练样本总数，y_i 和 $\tilde{y}_i$ 分别是第 i 条数据的目标标签和预测标签。

4.3. 分类。强化学习（RL）[98] 是机器学习的一个子领域，通过连续做出决策来解决问题 [99, 100]。近年来，强化学习在分类方面取得了出色的成果，因为它能够从噪声数据中学习有价值的特征或选取高层次样本。在 [101],中，分类问题被定义为一个序贯决策过程，通过利用多个因素来学习最优策略。然而，智能体与环境之间复杂的模拟在一定程度上增加了时间复杂度。另一项研究[102] 提出了一种从文本噪声数据中学习关系的解法。为此，所提出的模型分为两部分：实例选择器和关系分类器。实例选择器旨在借助智能体从噪声数据中提取高质量句子。同时，关系分类器从选出的干净数据中学习，以提升性能，并向实例选择器提供延迟奖励反馈。最终，该模型能够得到更好的分类结果和更高质量的数据集。[103–106] 中的作者考虑使用深度强化学习来学习有用的训练数据特征。总体而言，他们提升了分类器的有价值特征。[107]中的研究使用强化学习对时间序列数据进行分类，并设计了奖励函数和马尔可夫模型。迄今为止，关于不平衡数据分类的研究仍然很少，尤其是利用强化学习处理自然语言方面。在 [108],中，研究者提出了一种集成剪枝方法，在强化学习框架下选出最佳子分类器。该方法适用于小规模数据，因为实际上不可能选择具有许多子类别的分类器。

本节介绍如何应用强化学习来避免不平衡分类。总体而言，智能体在每个步骤都会接收一个样本并对其进行分类。之后，环境会向智能体提供即时奖励和后续奖励。当智能体正确分类样本时，环境会给予其正奖励；否则，智能体会收到负奖励。最后，智能体通过最大化累计奖励来学习最优行为，进而尽可能准确地对样本进行分类。

设 $D = \{x_1\ l_1,\ x_2\ l_2,\ \ldots,\ x_T\ l_T\}$，$(\ )(\ )\ldots(\ )$ 为训练数据，其中 $x_i = (q_i,\ a_i)$ 是第 i 个样本，因此 q_i 和 a_i 分别是输入模型的第i个问题和答案。$l_i \in \{0,\ 1\}$ 展示了第 i个示例的目标。我们考虑智能体的以下条件。

4.3.1. 策略 π_θ。策略 π_θ 是一个映射函数π：$S \longrightarrow A$，其中 $\pi_\theta(s_t)$ 表示智能体在状态 s_t 下执行的动作 a_t。在我们的工作中，所提出的带有集合权重 θ 的分类被视为策略 π_θ。

4.3.2. 状态 s_t。训练数据集中的每个示例都被描述为一个状态。在训练开始时，智能体将第一个数据 x_1 作为初始状态 s_1。每个时间步 t 的状态 s_t 对应于训练数据集中的 x_t。每次迭代中，智能体所处理的样本顺序都不同。

4.3.3. 动作 a_t. 智能体执行的动作是预测类别标签。因此，智能体的表现与训练数据集的标签有关。建议使用二元分类器，$a_t \in \{0,1\}$，其中 0 和 1 分别表示少数类和多数类。在此语境下，相关问答的标签为 1，不相关问答的标签为 0。

4.3.4. 奖励 r_t. 如果样本分类正确，智能体会获得正分；否则会获得负分。由于少数类实例数量较少、因而更为关键，算法应更加重视少数类分数的大小。奖励函数描述如下：

$$r(s_t, a_t, l_t) = \begin{cases} +1, a_t = l_t \text{ and } s_t \in D_P \\ -1, a_t \neq l_t \text{ and } s_t \in D_P \\ \lambda, a_t = l_t \text{ and } s_t \in D_N \\ -\lambda, a_t \neq l_t \text{ and } s_t \in D_N \end{cases}, \tag{12}$$

其中，$\lambda \in [0,1]$、D_P 和 D_N 分别与少数类和多数类有关。l_t 是样本x_t的标签。奖励金额被视为预测该标签的成本。根据这一关系，当 $\lambda < 1$时，少数类的成本更高。如果所有类别的分布均衡，$\lambda = 1$，那么所有类别的预测成本相同。我们将在实验中考察 λ的不同取值。

4.3.5. 终止状态 E。回合是从初始状态到终止状态的转移轨迹 $(s_1,\ a_1,\ l_1)\{(s_2,\ a_2,\ l_2),\ 。\ \ldots,\ (s_t,\ a_t,\ l_t)$。当训练数据中的所有实例都完成分类，或智能体错误地将少数类实例分类时，一个回合结束。

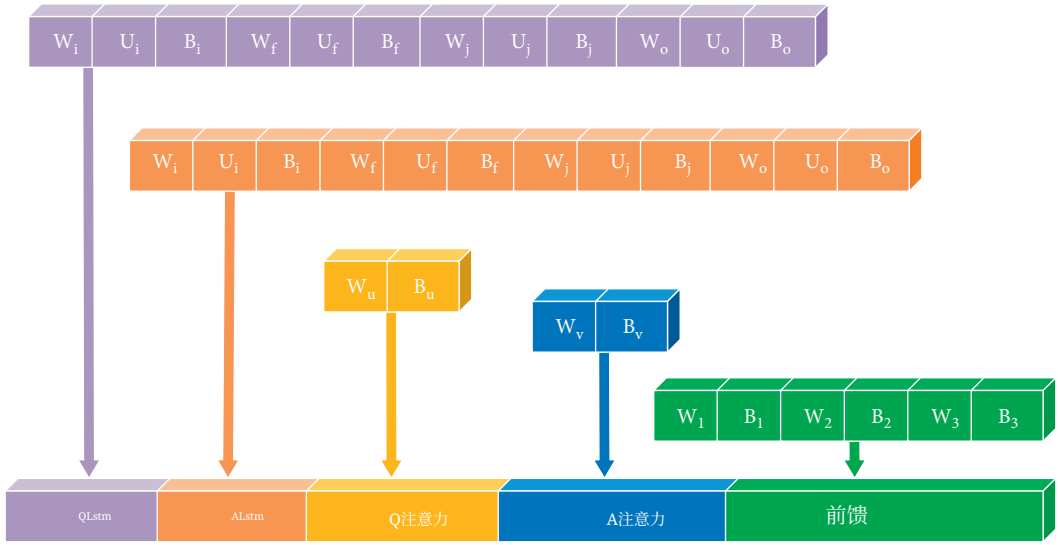


图3：向量中权重的放置位置。

4.3.6. 转移概率 P。模型转移概率，即 $p(s_{t+1}|s_t, a_t)$，是确定性的。智能体根据数据集中实例的顺序从状态 s_t 转移到状态 s_{t+1}。

在所提出的模型中，π 策略接收输入数据并计算其标签概率：

$$\pi(a \mid s) = P \cdot (a_t = a \mid s_t = s). \tag{13}$$

智能体旨在尽可能准确地识别数据输入样本。若智能体能够最大化其累积奖励，则可获得最佳性能，如下所示：

$$g_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}. \tag{14}$$

方程（14）称为回报函数，表示从时间 t 开始、折扣因子为$\gamma \in (0, 1]$ 时起，直至智能体在搜索空间中移动时所累积的总回报。强化学习中的动作价值 Q 表示在状态 s 下执行动作 a 的预期回报，可定义如下：

$$Q^{\pi}(s, a) = E_{\pi}[g_t \mid s_t = s, a_t = a]. \tag{15}$$

根据贝尔曼方程 [109]，方程（15）可以扩展为：

$$Q^{\pi}(s, a) = E_{\pi}[r_t + \gamma Q^{\pi}(s_{t+1}, a_{t+1}) \mid s_t = a, a_t = a]. \tag{16}$$

通过在策略 π 下最大化函数 Q，我们可以最大化累积奖励，即 Q^*。在函数 Q^* 下得到的最优策略 π^* 是最适用于我们模型的策略，如下所示：

$$\pi^*(a \mid s) = \begin{cases} 1, & a = \operatorname{argmax}_a Q^*(s, a) \\ 0, & \text{else} \end{cases}. \tag{17}$$

结合式（16）和式（17），函数 Q^* 的计算如下：

$$Q^*(s, a) = E_{\pi}[r_t + \gamma \max_a Q^*(s_{t+1}, a_{t+1}) | s_t = a, a_t = a]. \tag{18}$$

对于低维情况，将函数 Q 的值收集在表格中，并根据记录的值获取最优值。然而，当问题的维度是连续的时，函数 Q 便无法再求解。为解决这一问题，采用深度 Q学习算法，通过深度神经网络对函数 Q 建模。为此，将由式（18）得到的元组（s，a，r，s'）存储在回放记忆 M中。智能体从 M 中随机选取一个包含 B 条状态转移的小批量，并根据以下损失函数对深度 Q 网络执行梯度下降算法：

$$L(\theta_k) = \sum_{(s,a,r,s') \in B} (y - Q(s, a; \theta_k))^2, \tag{19}$$

其中，y 是函数 Q 的预测值，其公式如下：

$$y = \begin{cases} r, & \text{end} = \text{True} \\ r + \gamma \max_{a'} Q(s', a'; \theta_{k-1}), & \text{end} = \text{True} \end{cases}, \tag{20}$$

其中，s' 表示下一个状态 s，a' 则是在状态 s'下执行的动作。

4.4. 总体算法。我们根据上述定义的内容设计仿真环境。策略的网络架构在很大程度上取决于训练样例的复杂度和数量。在此背景下，网络的输入取决于训练

样本，输出等于实例数据类别的数量。算法2展示了该模型的一般训练算法。首先，使用人工蜂群算法初始化策略 π 的初始权重，然后智能体持续进行训练，直到达到最优策略。动作的选择基于贪心策略，

所选动作由算法 3 进行评估。算法重复 E 次，其中本文将 E 设为 15,000。每一步都会存储策略网络权重。

5. 结果

*5.1.*数据集。 包含大量负样本对的数据集，可能是评估所提出系统性能的最佳选择之一。我们在三个数据集 LegalQA、TrecQA 和 WikiQA 上开展实验，这些数据集受到许多研究人员的广泛关注。三个数据集中的负样本对均多于正样本对。所有数据集的统计信息见表 1：

(i) TrecQA [110] 源自 TREC 赛道数据。Yao 等人 [10] 构建了完整的正、负样本对集合。该数据库提供两个训练数据集：TRAIN 和 TRAIN-ALL。TRAIN-ALL 中答案的正确性通过正则表达式匹配样本对自动检查。TRAIN、DEV 和 TEST 数据中的所有答案均经过人工判定。我们使用 TRAIN-ALL 数据训练模型。

(ii) LegalQA [111] 是一个中文数据集，收录了从中国某协会收集的法律咨询问题。持证律师已回答用户在线提出的问题。LegalQA 包含四个字段：问题主题、问题正文、答案和标签。正样本对直接在线提供，并作为真实标注。

（iii）WikiQA [112] 是一个开放域问答数据集，其中每个问题都关联到一个维基百科页面，并假定该页面是该问题的主题。为消除答案句偏差，页面摘要部分中的所有答案都被视为候选答案。

5.2. 评估指标。根据先前的研究，MAP 和 MRR 是评估答案选择任务 [77]最常用的标准。MAP 衡量对答案进行排序以返回相应答案的能力。然而，当找到高分匹配项时，MRR 会重复计算：

（i）MAP（平均准确率均值）按如下方式计算排序结果的平均准确率均值：

$$\mathrm{MAP}(Q) = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{n_i}\sum_{j=1}^{n_i}\mathrm{precision}\left(R_{ij}\right), \tag{21}$$

其中，Q 表示问题集合，n_i 表示第 i 个问题的答案数量，R_{ij} 表示问题 j 的排序结果集合，从最佳结果到第 j 个答案依次排列。

(ii) 平均倒数排名（mean reciprocal rank，MRR）根据第一个正确答案的位置评估模型的适用性，计算公式如下：

$$\mathrm{MRR}(Q) = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{r_i}, \tag{22}$$

其中，r_i 表示第 i 个问题中第一个匹配答案的位置。

5.3. 基线方法。我们采用几种最先进的方法对我们的 RLAS-BIABC 模型进行答案选择评估。以下是这些方法的详细信息：

KABLSTM [113]是一种基于注意力双向长短期记忆网络的知识感知方法。该方法使用知识图谱（KG）学习问题和答案的表示。

EATS [75] 采用循环神经网络来衡量问答对之间的相似度。首先，它将每个命名实体替换为一个特定词语。该系统通过注意力机制计算句子表示向量。最后，将这些向量输入前馈网络，并通过最后一层的 Sigmoid 函数计算相似度。

AM-BLSTM [114] 分别为问题和答案使用两个长短期记忆网络。所得嵌入向量经过组合后，输入多层感知机（MLP）网络进行分类。此外，还采用了针对各类别的惩罚项等传统技术，以防止不平衡分类。

BERT-Base [115] 提出了一种用于答案选择的搜索引擎和 Transformer 模型方法。本文采用杰卡德相似度和比较-聚合等简单模型，对问题的答案进行排序。

DRCN [116] 提出了一种基于密集连接循环网络和协同注意力网络的架构，并在顶层保留隐藏特征。本文使用注意力机制执行连接操作，以更好地保留信息。此外，还采用自编码器来减少信息量。

P-CNN [117] 提出了一种使用位置卷积神经网络进行文本匹配的新方法，该方法考虑了词语、短语和句子层面的位置信息。

DARCNN [118]结合了双向长短期记忆网络、自注意力、交叉注意力和卷积神经网络，以提取问题和候选答案的全局和局部特征，从而实现更好的语义建模。最后，该模型使用多层感知机网络为问答对分配分数。

DASL [119] 提出了一种采用贝叶斯神经网络（BNN）的模型，以有效优化排序学习过程中的损失。本文的另一项研究内容是如何结合主动学习和自定步调学习来训练模型。

KAAS [120] 采用了一个用于答案选择的交互式知识增强注意力网络，该网络能够从多个层次提取丰富的问答知识特征。此外，还采用了注意力网络和自注意力网络来学习句子的语义特征。

输入： D ← ({ x_1 l_1 , x_2 l_2 , , x_T l_T },) (,) ..... (,) : 大小为 T 的训练数据集

使用算法1初始化策略 π 的权重
初始化环境 ε
初始化回放记忆 M
对于回合 e ← 1 到 E**执行**
  打乱数据集 D
  s_1 ← x_1
  对于t= 1 到 T**执行**
    a_t ← $\pi_0(s_t)$ //根据 ε-贪心选择一个动作
    $[r_t, \text{结束}_t]$ ← 奖励(x_t, a_t, l_t)
    $s_{t+1} = x_{t+1}$
    保存 $(s_t, a_t, r_1, s_{t+1}, \text{结束}_t)$ 到 M
    从 M 中随机抽取一个由状态转移组成的小批量（s_k， a_k， r_k， s_{k+1}）
    $y_k = \begin{cases} r_k, & \text{end}_k \leftarrow \text{真} \\ r_k + \gamma \text{最大}_{a'} Q(s_{k+1}, a'; \theta), & \text{end}_k \leftarrow \text{假} \end{cases}$
    计算关于 θ 的梯度： $d\theta$ ← $d\theta + \partial(y_k - Q(s, a; \theta))^2/\partial\theta$
    如果end$_t$ ← 真**则**
      中断
    结束如果
  结束循环
结束循环

Algorithm 2：RIAS-BIABC 训练伪代码。

奖励函数 (x_t, a_t, l_t):
  end_t ← 假
  if$x_t \in D_p$**then**
    if$a_t == l_t$**then**
      $r_t = 1$
    else
      $r_t = -1$
      end_t ← 真
    结束如果
  否则:
    if$a_t == l_t$**then**
      $r_t = \lambda$
    否则:
      $r_t = -\lambda$
    结束如果
  结束如果
  返回$[r_t, \text{结束}_t]$

Algorithm 3：奖励函数的伪代码。

表 1：LegalQA、TrecQA 和 WikiQA 数据集的统计信息。

数据集（训练集/开发集/测试集）	# 问题	# 问答对	% 正确率
LegalQA	10,526/1,593/3,035	100,590/11,965/26,913	21.8/24.4/22.9
TrecQA	1,229/82/100	53,417/1,148/1,517	12.0/19.3/18.7
WikiQA	873/126/243	20,360/1,130/2,352	12.0/12.4/12.5

“正确率”表示匹配问答对的比例。

5.4. 实现细节。 本研究使用 Python 和 PyTorch 进行实现，并使用 Jupyter 编写项目代码。本研究还使用了另一个库 NLTK。该库提供了用于处理 Python 自然语言的类和方法。本

该库能够执行广泛的自然语言处理操作。我们使用双层双向长短期记忆网络。此外，由于两个网络中的向量相互连接，我们在数据进入前馈神经网络之前进行批归一化。表2列出了其他参数的值。

我们的项目使用64位Windows操作系统、64GB随机存取存储器和图形处理器。经过50、60和100个训练轮次后，分别获得了LegalQA、TrecQA和WikiQA的最佳模型。整个训练过程在这三个数据集上分别耗时5、20和60小时。

5.5. 实验结果。由于启发式算法具有随机性，我们将所有实验重复进行了 10 次。三个数据集的定量结果见表3。除了将所提出的方法与最先进算法进行比较外，为了评估人工蜂群算法和强化学习组件对模型的有效性，我们还采用了三种技术：AS +随机权重、AS-BIABC 和 RLAS。AS +随机权重是一种仅使用随机权重进行初始权重设置的系统。AS-BIABC 和 RLAS 模型分别仅采用人工蜂群算法和强化学习。对于 LegalQA 数据集，RLAS-BIABC 模型在平均准确率均值和平均倒数排名指标上均优于包括 IKAAS 在内的其他模型，因此，我们的模型在这两项指标上的误差分别降低了 40% 以上和 24% 以上。将 RLAS-BIABC 与 AS-BIABC 和 RLAS 进行比较可知，其误差率降低了约 51%，这表明初始化和强化学习方法的重要性。对于 TrecQA 数据集，我们的算法取得了最高的平均准确率均值和平均倒数排名，其次是 EAT 算法。在该数据库中，平均准确率均值和平均倒数排名指标的误差改善率分别约为 30.13% 和 21.00%。在 WikiQA 数据集中，与 IKAAS 和 DRCN 相比，RLAS-BIABC 的分类错误分别降低了 32% 以上和 42% 以上。

接下来，我们证明改进的人工蜂群算法比其他算法更强大。为进行公平比较，我们固定算法的所有组成部分，包括长短期记忆网络、注意力机制和强化学习，只更换训练器。为此，我们将所提出的训练器与六种传统算法进行比较，包括 GDM [121],GDA [122], GDMA [123], OSS [124], 和 BR[125], , 以及八种元启发式算法，包括灰狼优化算法[126], 蝙蝠算法 [127],蜻蜓算法 [128], SSA [129], COA [130],HMS[131], WOA [132], 和人工蜂群算法 [133]。在所有元启发式方法中，种群规模和函数评估次数分别为 100 和 3,000。算法的其余参数见表4。元启发式算法和传统算法的结果汇总于表5。对于传统算法和元启发式算法，RLAS-AM-BR 和 RLAS-BABC 在所有数据集上均表现最佳。正如预期，元启发式算法的表现优于传统算法。毫不夸张地说，改进的人工蜂群算法的表现优于所有其他算法；与表现最佳的算法，即 ABC 主版本相比，它可将误差降低约 16%。

5.5.1. 多数类奖励值的影响。环境通过考虑奖励函数帮助智能体实现目标。本文为少数类和多数类设置了两种不同的奖励。少数类奖励设为 $+1/-1$ ，而多数类奖励设为 $+\lambda/-\lambda$。为了研究λ 的值对所提出模型的影响，我们使用集合中的值对其进行测试

表2：模型的参数。

参数	值
批次大小	128
嵌入维度	60
最大句子长度	80
激活函数（长短期记忆网络和全连接）	ReLU
全连接隐藏层	8

$0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1\{\}$。三个数据集上的实验结果如图4所示。可以看到，对于 LegalQA 数据集，当 λ 的值处于 $[0,0.4]$范围内时，呈上升趋势；而当值为 $(0.4,1]$ 时，则呈下降趋势。因此，我们将该数据集的 λ 值固定为 0.5。TrecQA 和 WikiQA 数据集的 λ 最佳值均为 0.5。一般而言，随着数据集规模增大，负样本对的数量也会增加，因此 λ趋于减小。当 $\lambda \to 0$ 时，多数类的重要性会被忽视；而对于大多数 $\lambda \to 1$ 的情况，两个类别的重要性相同。

5.5.2. 损失函数探索。传统技术，包括调整损失函数和数据增强，也可以应对数据不平衡问题。然而，这些技术在很大程度上取决于具体问题。同时，损失函数的作用更加多样，因为它可以提高少数类的显著性。为检验各损失函数在我们的模型上的效果，我们选取了五种损失函数：加权交叉熵（WCE） [134],平衡交叉熵（BCE）[135], 焦点损失（FL） [136], Dice损失（DL） [137], 和Tversky损失（TL） [138]。WCE和BCE损失函数会为正样本和负样本赋予权重。FL函数适用于存在不平衡数据的应用场景。它会降低简单样本的贡献，使模型能够更加专注于学习复杂样本 [139]。表6展示了这些损失函数在三个数据集上的评估结果。结果表明，所有函数在这三个数据集上的MRR和MAP大致相同。正如预期，FL函数的表现优于其他函数，因此其性能比采用常规损失函数的算法（即RLAS-BABC模型）高出约51.16%。

5.5.3. 案例研究。本节旨在定性评估强化学习在我们模型中的有效性。为此，我们从 TrecQA数据集中随机选取一个样本。对于问题“诺贝尔奖首次颁发于何时？”，表7列出了排名靠前的答案。左列展示了未使用强化学习时的模型结果，右列展示了使用强化学习时的模型结果。结果表明，未使用强化学习的模型更倾向于给负面回答较高的分数。然而，使用强化学习的模型则尽可能多地为该问题的答案分配了分数。

表3：所提出模型与其他模型的评估结果。

方法	数据集					
	LegalQA		TrecQA		WikiQA	
	MAP	MRR	MAP	MRR	MAP	MRR
KABLSTM [113]	0.751	0.790	0.792†	0.844†	0.732†	0.749†
EATS [75]	0.778	0.810	0.854†	0.881†	0.700†	0.715†
AM-BLSTM [114]	0.786	0.836	0.818	0.827	0.780	0.788
BERT-Base [115]	0.838	0.850	0.823	0.812	0.813†	0.828†
DRCN [116]	0.828	0.859	0.802	0.832	0.804†	0.862†
P-CNN [117]	0.715	0.729	0.680	0.698	0.734†	0.737†
DARCNN [118]	0.700	0.745	0.743	0.725	0.734†	0.750†
DASL [119]	0.804	0.816	0.824	0.831	0.768	0.795
IKAAS [120]	0.825†	0.883†	0.823†	0.868†	0.835	0.849
答案选择 +随机权重	0.758 ± 0.000	0.801 ± 0.001	0.796 ± 0.000	0.806 ± 0.002	0.771 ± 0.002	0.792 ± 0.009
AS-BIABC	0.788 ± 0.012	0.815 ± 0.008	0.802 ± 0.005	0.826 ± 0.002	0.803 ± 0.000	0.845 ± 0.025
RLAS	0.855 ± 0.102	0.872 ± 0.018	0.862 ± 0.014	0.883 ± 0.150	0.852 ± 0.025	0.876 ± 0.026
RLAS-BIABC	0.895 ± 0.020	0.912 ± 0.001	0.898 ± 0.015	0.906 ± 0.092	0.888 ± 0.036	0.891 ± 0.017

† 表示结果取自相关论文。

表4：实验的参数设置。

算法	参数	值
ABC	限制	$n_e \times$问题维数
	n_o	群体的 50%
	n_e	群体的 50%
	n_s	1
GWO	无参数	
BAT	响度更新常数	0.4
	发射率更新常数	0.6
	初始脉冲发射率	0.002
DA	缩放因子	0.3
	交叉概率	0.7
SSA	无参数	
COA	外来解发现率	
	簇数	5
HMS	最少思维过程数	2
	最多思维过程数	5
WOA	C	1
	B	1

表5：其他方法的初始化性能。

方法	数据集					
	LegalQA		TrecQA		WikiQA	
	MAP	MRR	MAP	MRR	MAP	MRR
RLAS-BGDM	0.796 ± 0.002	0.819 ± 0.026	0.824 ± 0.093	0.836 ± 0.026	0.810 ± 0.056	0.825 ± 0.136
RLAS-BGDA	0.783 ± 0.125	0.776 ± 0.095	0.769 ± 0.025	0.786 ± 0.269	0.745 ± 0.136	0.761 ± 0.002
RLAS-BGDMA	0.791 ± 0.005	0.772 ± 0.103	0.796 ± 0.126	0.812 ± 0.236	0.793 ± 0.026	0.793 ± 0.005
RLAS-BOSS	0.810 ± 0.136	0.814 ± 0.004	0.853 ± 0.023	0.863 ± 0.026	0.840 ± 0.027	0.855 ± 0.127
RLAS-BBR	0.842 ± 0.009	0.853 ± 0.000	0.860 ± 0.036	0.878 ± 0.120	0.852 ± 0.103	0.870 ± 0.035
RLAS-BGWO	0.771 ± 0.205	0.783 ± 0.018	0.755 ± 0.072	0.781 ± 0.126	0.755 ± 0.025	0.773 ± 0.026
RLAS-BBAT	0.862 ± 0.003	0.818 ± 0.019	0.876 ± 0.093	0.880 ± 0.239	0.852 ± 0.061	0.873 ± 0.082
RLAS-BDA	0.816 ± 0.072	0.829 ± 0.022	0.863 ± 0.002	0.883 ± 0.056	0.836 ± 0.082	0.862 ± 0.091
RLAS-BSSA	0.747 ± 0.029	0.769 ± 0.072	0.750 ± 0.042	0.763 ± 0.025	0.746 ± 0.041	0.755 ± 0.001
RLAS-BCOA	0.860 ± 0.085	0.889 ± 0.089	0.882 ± 0.063	0.897 ± 0.237	0.872 ± 0.093	0.862 ± 0.017
RLAS-BHMS	0.849 ± 0.002	0.880 ± 0.123	0.879 ± 0.090	0.893 ± 0.036	0.840 ± 0.100	0.870 ± 0.009
RLAS-BGDM	0.752 ± 0.012	0.753 ± 0.027	0.769 ± 0.058	0.789 ± 0.085	0.731 ± 0.000	0.760 ± 0.018
RLAS-BABC	0.875 ± 0.004	0.906 ± 0.021	0.888 ± 0.046	0.900 ± 0.082	0.878 ± 0.016	0.889 ± 0.023

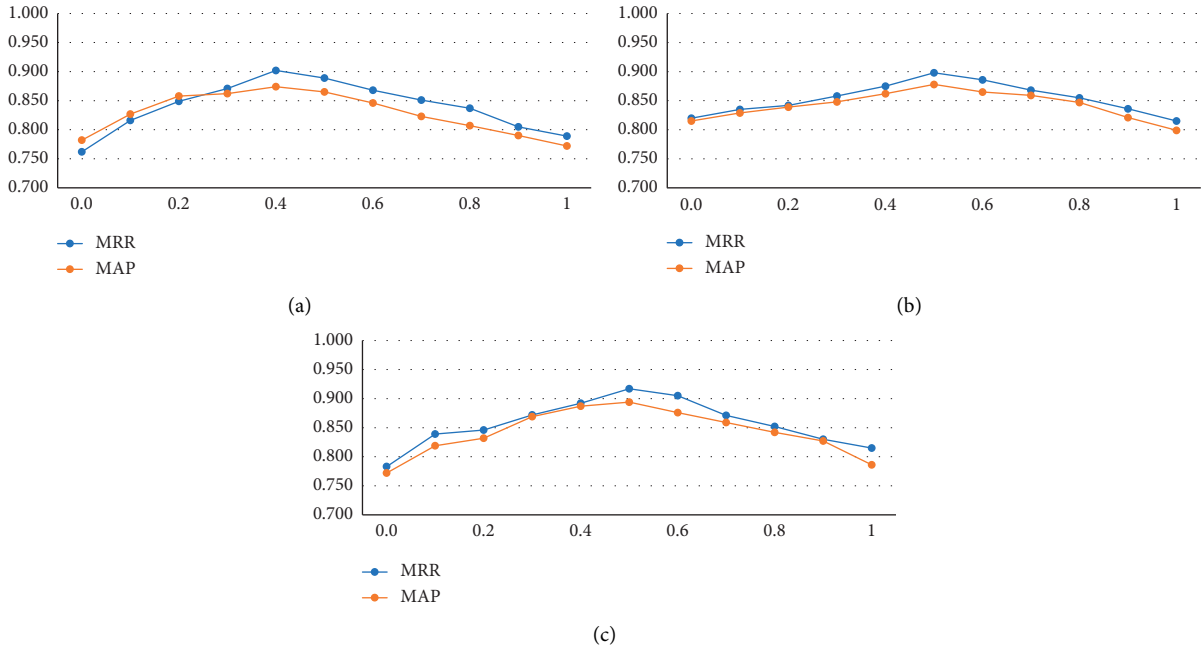


图4：通过修改三个数据集的 λ 值来改变标准的过程：(a) LegalQA 数据集；(b) TrecQA 数据集；(c) WikiQA 数据集。

表6：模型采用不同损失函数的结果。

模型	数据集					
	LegalQA		TrecQA		WikiQA	
	MAP	MRR	MAP	MRR	MAP	MRR
AS-BIABC + WCE	0.781 ± 0.002	0.819 ± 0.026	0.772 ± 0.005	0.780 ± 0.145	0.795 ± 0.010	0.792 ± 0.012
AS-BIABC + 二元交叉熵	0.789 ± 0.000	0.812 ± 0.120	0.786 ± 0.073	0.804 ± 0.025	0.783 ± 0.074	0.814 ± 0.002
AS-BIABC + 焦点损失	0.842 ± 0.048	0.838 ± 0.056	0.839 ± 0.090	0.829 ± 0.012	0.832 ± 0.005	0.822 ± 0.006
AS-BIABC + Dice损失	0.838 ± 0.089	0.808 ± 0.135	0.810 ± 0.074	0.770 ± 0.203	0.806 ± 0.082	0.804 ± 0.120
AS-BIABC + Tversky 损失	0.785 ± 0.096	0.783 ± 0.582	0.821 ± 0.006	0.800 ± 0.041	0.823 ± 0.018	0.799 ± 0.005

表7：对于“诺贝尔奖首次颁发于何时？”这个问题，表中展示了模型在采用和未采用强化学习时排名前五的答案。

Rank	未采用强化学习时排名靠前的答案	采用强化学习后排名靠前的答案
1	首次颁奖典礼于**1901年**举行	授予无国界医生的奖项令人想起首届诺贝尔世纪和平奖颁发于**1901年**，其创始人红十字会是共同获奖者之一
2	由五名成员组成的奖项委员会在其每年五六次会议期间严格保密，并拒绝评论或公布候选人的姓名	奖项于**1901年**首次颁发，颁奖日期始终为12月10日，诺贝尔逝世周年纪念日
3	**1901年**，瑞典颁发了首届诺贝尔奖，以表彰在医学领域的柏林人埃米尔·冯·贝林，因他研制出治疗白喉的血清	其中包括首届奖项的得主**1901年**，苏利·普吕多姆
4	奖项于**1901年**首次颁发，总是在12月诺贝尔逝世10周年	首届颁奖典礼于**1901年**举行
5	“我们都知道，仍有许多重大问题亟待解决，”颁奖委员会主席弗朗西斯·塞耶斯泰德说	例如，公告发布后的第二天，评论家诺曼霍姆斯·皮尔逊抱怨说，这位女性赛珍珠竟然获得了诺贝尔奖文学奖

“在1901年”是真实标注答案，斜体文字是在问题中出现的术语。

表 8：不同词嵌入在该模型上的结果。

词嵌入	数据集					
	LegalQA		TrecQA		WikiQA	
	MAP	MRR	MAP	MRR	MAP	MRR
独热编码	0.679 ± 0.042	0.569 ± 0.002	0.711 ± 0.120	0.653 ± 0.081	0.649 ± 0.089	0.589 ± 0.093
CBOW	0.869 ± 0.006	0.843 ± 0.000	0.889 ± 0.078	0.869 ± 0.120	0.836 ± 0.012	0.828 ± 0.010
Skip-gram	0.874 ± 0.052	0.872 ± 0.075	0.878 ± 0.030	0.858 ± 0.002	0.847 ± 0.014	0.853 ± 0.014
GloVe	0.812 ± 0.027	0.853 ± 0.082	0.795 ± 0.140	0.821 ± 0.074	0.782 ± 0.039	0.806 ± 0.009
FastText	0.881 ± 0.002	0.901 ± 0.041	0.886 ± 0.093	0.876 ± 0.002	0.861 ± 0.099	0.870 ± 0.000

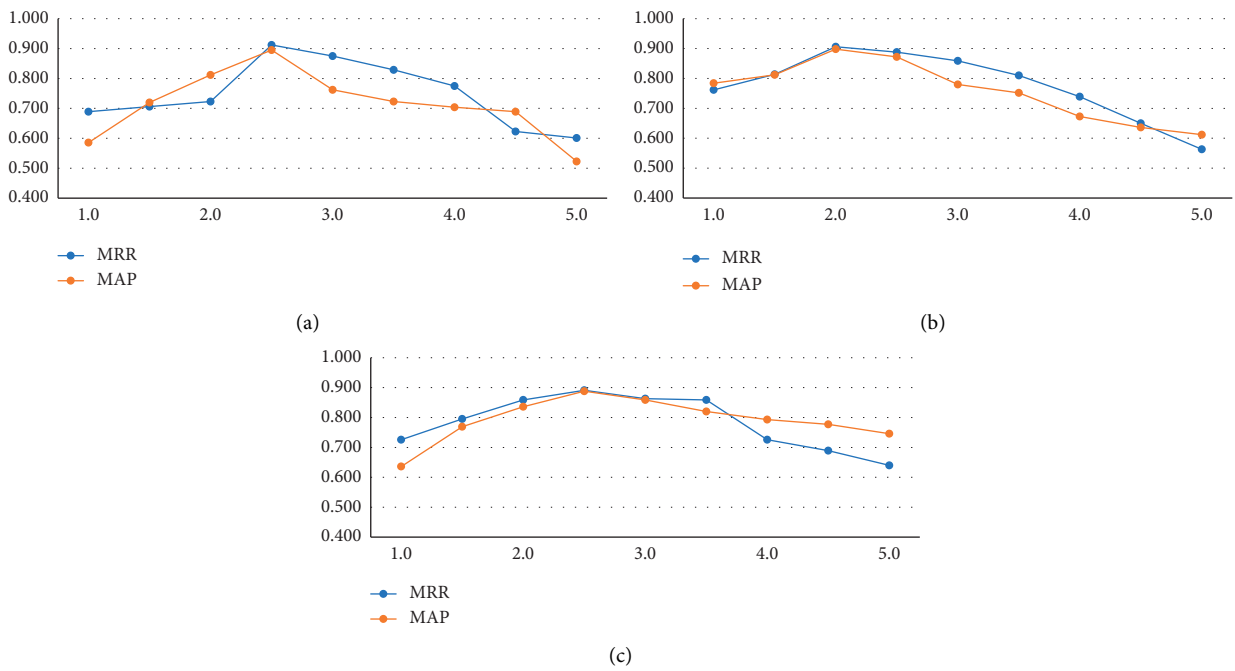


图5：通过修改三个数据集的 F值来改变标准的过程：(a) LegalQA 数据集；(b) TrecQA数据集；(c) WikiQA 数据集。

5.5.4. 词嵌入探索。词嵌入是深度学习模型的主要组成部分之一，因为输入会被解释为向量；如果嵌入不正确，模型可能会受到误导。本研究采用 BERT 模型作为词嵌入，该模型是最新的嵌入模型之一。为了在我们的模型上检验其他词嵌入，我们采用了四种词嵌入：独热编码 [140], 连续词袋模型 [141], Skip-gram[93],GloVe [94], 和 FastText [95]。独热编码是将分类数据变量转换为可供深度学习算法使用的形式的关键过程，能够提高预测和分类准确率。这种词嵌入为每个类别创建一个新的二进制特征，并将值 1 分配给每个样本中与其原始类别对应的特征。CBOW 和 Skip-gram 是使用神经网络将词映射为其嵌入向量的模型。GloVe 词嵌入是一种无监督学习算法，基于语料库中汇总的全局词-词共现统计信息实现。FastText 是一种词嵌入，是对 Skip-gram 模型的扩展。该方法不再为词语学习向量，而是将每个词表示为一个n-gram 的字符。本次实验的结果如表 8 所示。正如预期，独热编码在所有词嵌入中表现最差；因此，在这种词嵌入表现最佳的 TrecQA 数据集上，平均准确率均值和平均倒数排名指标的提升率分别约为 64.70% 和 72.91%。由于架构相似，CBOW 和 Skip-gram 在三个数据集上的表现几乎相同，且均优于 GloVe 词嵌入。FastText 对所有模型而言都是最佳词嵌入，但在 BERT 模型上的表现仍然不佳。与 FastText 模型相比，BERT 模型在 WikiQA、TrecQA 和 LegalQA 数据集上的误差分别降低了 11%、10% 和 19% 以上。

5.5.5. 参数F对模型的影响。为考察式（10）中参数F对所提出方法的算法性能的影响，我们将F设为 0.5、1、1.5、2、2.5、3.5、4、4.5 和 5。这些设置在三个数据集上得到的结果如图5所示。可以看出，对于 LegalQA 数据集，当F从 0 增至 2 时，

算法的表现越来越好。然而，可以观察到，当F 从 2 增至 5 时，方法的性能会下降。这意味着，F 的值过小或过大都会削弱算法性能。对于 TrecQA 和 WikiQA 数据集，F 等于 1.5 和 2 时，算法的性能优于其他取值。

6. 结论与未来工作

本文提出了一种用于答案选择（AS）的方法，称为 RLAS-BIABC。该方法建立在基于注意力机制的长短期记忆网络（LSTM）方法和 BERT 词嵌入之上，并结合改进的人工蜂群算法进行预训练，以及通过强化学习训练反向传播算法。RLAS-AM-ABC 模型旨在对正类和负类进行分类，其中正样本对包含一个问题和一个真实答案。相比之下，负样本对包含一个问题和一个虚假答案。由于数据集中存在大量负样本对，RLAS-BIABC 被转化为不平衡分类问题。为解决这一问题，我们将模型构建为一个序贯决策过程。在此过程中，环境会在每一步为每个分类行为分配奖励，其中少数类获得更高的奖励。该过程会持续进行，直到智能体错误地将少数类样本分类，或回合数结束。初始权重设置是深度模型的另一项重要特性，可能导致模型陷入局部最优。为解决这一问题，我们使用改进的人工蜂群算法初始化策略权重。本文提出了一种互学习技术：根据互学习因子，在两个个体中选择适应度更高的个体，并利用其修改生成的候选食物源。我们设计了实验，以考察影响模型的因素。分析结果表明，强化学习、BERT 和改进的人工蜂群算法在答案选择方面具有较强的能力。

在未来的工作中，我们将在改进所提出模型的同时，尝试考察所提出的分类器在其他自然语言处理应用中的有效性。另一项任务是提供一个能够生成问题答案的模型。作为一种解决方案，我们将重点关注生成对抗网络，这种技术如今已广泛应用于几乎所有领域，包括自然语言处理任务。

数据可用性

用于支持本研究发现的数据已包含在本文中。我们在文章中提供了数据集的相关信息（见第 5.1 节）。

利益冲突

作者声明不存在利益冲突。

参考文献

[1]Y. Sun, C. Chen, T. Xia, 和 X. Zhao，“QuGAN：用于藏语问题解答语料库生成的准生成对抗网络”，*IEEE Access*，第 7 卷，第 116247–116255 页，2019年。[2]J. Huang，“一种用于答案选择的结合注意力机制的多尺寸神经网络”，2021年，https://arxiv.org/abs/2105.03278。

[3] W. Lei、Y. Xiang、Y. Wang、Q. Zhong、M. Liu 和 M.-Y. Kan，“语言特性对于隐式篇章关系识别至关重要：结合语义交互、主题连贯性与归因”，载于 *AAAI*人工智能会议论文集，第 32 卷，第 1 页，路易斯安那州，美国，2018年2月。[4]S. Li、W. Lei、Q. Wu、X. He、P. Jiang 和 T.-S. Chua，“无缝统一属性与项目：面向冷启动用户的对话式推荐”，2020年，https://arxiv. org/abs/2005.12979。[5] S. Zhang、X. Zhang、H. Wang、J. Cheng、P. Li 和 Z. Ding，“使用端到端字符级多尺度卷积神经网络进行中文医疗问答匹配”，*Applied Sciences*，第 7 卷，第 8 期，第 767 页，2017年。[6] H. Yu、M. Lee、D. Kaufman 等，“面向医生的定义型问答系统的开发、实施与认知评估”，生物医学信息学杂志，第 40 卷，第 3 期，第 236–251 页，2007年。[7]X. Xu、L. He、A. Shimada、R.-i. Taniguchi 和 H. Lu，“通过潜在语义哈希学习用于跨模态检索的统一二进制码”，*Neurocomputing*，第 213 卷，第 191–203 页，2016年。[8] X. Xu、F. Shen、Y. Yang、H. T. Shen 和 X. Li，“学习用于大规模跨模态检索的判别式二进制码”，*IEEE Transactions on Image Processing*，第 26 卷，第 5 期，第 2494–2507 页，2017年。[9] S. Jain 和 T. Dodiya，“医疗问答系统的基于规则的架构”，载于第二届问题求解软计算国际会议论文集（*SocProS 2012*），第 1225–1233 页，施普林格，斋浦尔，印度，2012年12月。[10] X. Yao、B. Van Durme、C. Callison-Burch 和 P. Clark，“将答案抽取作为结合树编辑距离的序列标注”，载于*2013*年计算语言学协会北美分会会议论文集：人类语言技术，第 858–867 页，亚特兰大，佐治亚州，2013年6月。[11] S.-J. Yen、Y.-C. Wu、J.-C. Yang、Y.-S. Lee、C.-J. Lee 和 J.-J. Liu，“基于支持向量机的问答上下文排序模型”，*InformationSciences*，第 224 卷，第 77–87 页，2013年。[12] B. L. Cairns、R. D. Nielsen、J. J. Masanz 等，“MiPACQ临床问答系统”，载于*AMIA*……年度研讨会论文集。*AMIA*研讨会，第 2011 卷，第 171–80 页，美国医学信息学会，华盛顿特区，美国，2011年10月。[13] W. Lu、H. Wu、P. Jian、Y. Huang 和 H. Huang，“基于分类器组合的词义消歧实证研究”，*IEICE* - 信息与系统汇刊，第 E101.D 卷，第 1 期，第 225–233 页，2018年。[14] S. Wang 和 L. Cao，“通过学习显式和隐藏的项目依赖关系推断隐式规则”，*IEEE*系统、人类与控制论汇刊：系统，第 50 卷，第 3 期，第 935–946 页，2017年。[15] M. Heilman 和 N. A. Smith，“用于识别文本蕴含、释义及问题答案的树编辑模型”，载于人类语言技术：*2010*年计算语言学协会北美分会年会论文集，第 1011–1019 页，洛杉矶市中心，2010年6月。[16]H. Toba、Z.-Y. Ming、M. Adriani 和 T.-S. Chua，“利用分类器层级从社区问答档案中发现高质量答案”，*InformationSciences*，第 261 卷，第 101–115 页，2014年。

[17] S. Moravvej、M. Maleki Kahaki、M. Salimi Sartakhti 和 M. Joodaki，“基于生成对抗网络的高效抽取式摘要方法”，电气与计算机工程创新期刊，2021年。[18] S. V. Moravvej、A. Mirzaei 和 M. Safayani，“使用条件生成对抗网络（CGAN）的生物医学文本摘要”，2021年，https://arxiv.org/abs/2110.11870。[19] M. S. Sartakhti、M. J. M. Kahaki、S. V. Moravvej、M. javadi Joortani 和 A. Bagheri，“基于双向长短期记忆网络模型的COVID-19语料库波斯语语言模型”，载于*2021*年第五届模式识别与图像分析国际会议论文集（*IPRIA*），第 1–5 页，IEEE，卡尚，伊朗，2021年4月。[20]Z. Sobhaninia、H. Danesh、R. Kafieh、J. J. Balaji 和 V. Lakshminarayanan，“使用一种新型位置感知深度学习方法确定中央凹无血管区参数”，载于机器学习应用，第 2021 卷，第 1184311 页，国际光学与光子学学会，2021年。[21] Z. SobhaniNia、N. Karimi、P. Khadivi、R. Roshandel 和 S. Samavi，“使用中间残差编码器层进行脑肿瘤分类”，2020年，https://arxiv.org/abs/2011.00628。[22] Z. Sobhaninia、S. Rafiei、A. Emami 等，“使用多任务深度学习进行胎儿超声图像分割以测量生物测量参数”，载于*2019*年第*41*届*IEEE*医学与生物学工程学会年度国际会议论文集（*EMBC*），第 6545–6548 页，IEEE，柏林，德国，2019年7月。[23] J. Bao、D. Chen、F. Wen、H. Li 和 G. Hua，“CVAE-GAN：通过非对称训练生成细粒度图像”，载于*IEEE*国际计算机视觉会议论文集，第 2745–2754 页，意大利威尼斯，2017年10月。[24] P. Kim，“卷积神经网络”，载于*MATLAB*深度学习，第 121–147 页，Apress，伯克利，加利福尼亚州，2017年。[25] B. Hu、Z. Lu、H. Li 和 Q. Chen，“用于自然语言句子匹配的卷积神经网络架构”，神经信息处理系统进展，第 27 卷，2014年。[26] M. E. Peters、M. Neumann、M. Iyyer 等，“深度上下文化词表示”，2018年，https://arxiv.org/abs/ 1802.05365。[27] J. Devlin、M.-W. Chang、K. Lee 和 K. Toutanova，“BERT：用于语言理解的深度双向Transformer预训练”，2018年，https://arxiv.org/abs/1810.04805。[28] A. Vaswani、N. Shazeer、N. Parmar 等，“注意力机制就是你所需要的一切”，神经信息处理系统进展，第 30 卷，2017年。[29] J. Mozafari、A. Fatemi 和 M. A. Nematbakhsh，“BAS：一种使用BERT语言模型的答案选择方法”，2019年，https://arxiv.org/abs/1911.01528。[30] X.Xin Yao，“进化人工神经网络”，*IEEE*学报，第 87 卷，第 9 期，第 1423–1447 页，1999年。[31] D. E. Rumelhart、G. E. Hinton 和 R. J. Williams，“通过误差反向传播学习表示”，《自然》，第 323 卷，第 6088 期，第 533–536 页，1986年。[32] C. Blum 和 K. Socha，“使用蚁群优化训练前馈神经网络：一种模式分类应用”，载于第五届混合智能系统国际会议论文集（*HIS'05*），第 6 页，IEEE，里约热内卢，巴西，2005年11月。[33] R. S. Sexton、B. Alidaee、R. E. Dorsey 和 J. D. Johnson，“人工神经网络的全局优化：一种禁忌搜索应用”，*European Journal of Operational Re-search*，第106卷，第2-3期，第570–584页，1998年。[34] R. S. Sexton、R. E. Dorsey和J. D. Johnson，“神经网络优化：遗传算法与模拟退火的比较分析”，*European Journal ofOperational Research*，第114卷，第3期，第589–601页，1999年。[35] M. Dorigo、M. Birattari和T. Stutzle，“蚁群优化”，*IEEE Computational Intelligence Magazine*，第1卷，第4期，第28–39页，2006年。[36] F. Glover和M. Laguna，“禁忌搜索”，载于*Handbook ofCombinatorial Optimization*，第2093–2229页，施普林格，德国海德堡，1998年。[37] D. Bertsimas和J. Tsitsiklis，“模拟退火”，*Sta-tistical Science*，第8卷，第1期，第10–15页，1993年。[38] S. Mirjalili，“遗传算法，计算智能研究”，载于*Evolutionary Algorithms and Neural Net-works*，第43–55页，施普林格，德国海德堡，2019年。[39]C. A. de Sousa，“前馈神经网络权重初始化方法概述”，载于*2016*年国际神经网络联合会议（*IJCNN*），第52–59页，IEEE，加拿大不列颠哥伦比亚省温哥华，2016年7月。[40]A. Ranganathan，“Levenberg-Marquardt算法”，*LM*算法教程，第11卷，第1期，第101–110页，2004年。[41] C. Ozturk和D. Karaboga，“用于神经网络训练的混合人工蜂群算法”，载于*2011*年*IEEE*进化计算大会（*CEC*）论文集，第84–88页，IEEE，美国路易斯安那州新奥尔良，2011年7月。[42] S. V. Moravvej、S. J. Mousavirad、M. H. Moghadam和M. Saadatmand，“基于注意力机制和种群方法预训练参数、适用于不平衡类别的LSTM抄袭检测”，*NeuralInformation Processing*，施普林格国际出版公司，第13110卷，第690–701页，2021年。[43] Z. Abdi Alkareem Alyasseri、O. A. Alomari、M. A. Al-Betar等，“采用多目标布谷鸟搜索进行脑电图通道选择，以用于医疗保健应用中的人员识别保护系统”，计算智能与神经科学，第2022卷，文章编号5974634，共18页，2022年。[44] S. Vakilian、S. V. Moravvej和A. Fanian，“考虑雾层协作，利用布谷鸟算法优化雾节点的响应时间和能耗成本”，载于*2021*年第*5*届物联网应用国际会议论文集，第1–5页，IEEE，伊斯法罕大学，伊朗，2021年5月。[45] S. Vakilian、S. V. Moravvej和A. Fanian，“在物联网三层架构中使用人工蜂群（ABC）算法与雾节点协作”，载于*2021*年第*29*届伊朗电气工程会议（*ICEE*）论文集，第509–513页，IEEE，德黑兰，伊朗，2021年5月。[46] S. J. Mousavirad、G. Schaefer、I. Korovin、D. Oliva和Rde-Op，“RDE-OP：一种结合基于反向学习的区域差分进化算法，用于优化多层神经网络的学习过程”，载于进化计算应用国际会议论文集（*EvoStar*的一部分），第407–420页，施普林格，西班牙塞维利亚，2021年4月。[47] H. Han、W.-Y. Wang、B.-H. Mao和Borderline-Smote，“Borderline-SMOTE：一种用于不平衡数据集学习的新型过采样方法”，载于智能计算国际会议论文集，第878–887页，施普林格，中国合肥，2005年8月。[48] I. Mani和I. Zhang，“针对不平衡数据分布的k近邻方法：一项涉及信息抽取的案例研究”，

载于不平衡数据集学习研讨会论文集，第126卷，ICML，美国华盛顿特区，2003年8月。[49] S. Wang、W. Liu、J. Wu、L. Cao、Q. Meng和P. J. Kennedy，“在不平衡数据集上训练深度神经网络”，载于*2016*年国际神经网络联合会议（*IJCNN*）论文集，第4368–4374页，IEEE，加拿大不列颠哥伦比亚省温哥华，2016年7月。[50] C. Huang、Y. Li、C. C. Loy和X. Tang，“针对不平衡分类的深度表征学习”，载于*IEEE*计算机视觉与模式识别会议论文集，第5375–5384页，内华达州拉斯维加斯，2016年7月。[51] Y. Yan、M. Chen、M.-L. Shyu和S.-C. Chen，“不平衡多媒体数据分类的深度学习”，载于*2015*年*IEEE*多媒体国际研讨会（*ISM*）论文集，第483–488页，IEEE，美国佛罗里达州迈阿密，2015年12月。[52] S. H. Khan、M. Hayat、M. Bennamoun、F. A. Sohel和R. Togneri，“从不平衡数据中进行深度特征表示的代价敏感学习”，*IEEE Transactions onNeuralNetworks and Learning Systems*，第29卷，第8期，第3573–3587页，2017年。[53] Q. Dong、S. Gong和X. Zhu，“通过少数类增量校正实现不平衡深度学习”，*IEEE*模式分析与机器智能汇刊，第41卷，第6期，第1367–1381页，2018年。[54]Q. Dong、S. Gong和X. Zhu，“用于不平衡深度学习的类别校正困难样本挖掘”，载于*IEEE*国际计算机视觉会议论文集，第1851–1860页，意大利威尼斯，2017年10月。[55] V. Mnih、K. Kavukcuoglu、D. Silver等，“利用深度强化学习玩雅达利游戏”，2013年，https://arxiv.org/abs/ 1312.5602。[56]S. Gu、E. Holly、T. Lillicrap和S. Levine，“采用异步离策略更新的机器人操作深度强化学习”，载于*2017*年*IEEE*国际机器人与自动化会议（*ICRA*）论文集，第3389–3396页，IEEE，新加坡，2017年5月。[57] X. Zhao、L. Zhang、L. Xia、Z. Ding、D. Yin和J. Tang，“列表式推荐的深度强化学习”，2017年，https://arxiv.org/abs/1801.00209。[58] S. Wan、M. Dras、R. Dale和C. Paris，“利用基于依存关系的特征去除释义中的‘胡言乱语’”，载于澳大拉西亚语言技术研讨会论文集，第131–138页，澳大利亚悉尼，2006年11月。[59] G. A. Miller和WordNet，电子词汇数据库，麻省理工学院出版社，马萨诸塞州剑桥，1998年。[60] V. Punyakanok、D. Roth和W.-t. Yih，“通过依存树映射进行自然语言推理：在问答中的应用”，2004年，https://www.ideals.illinois.edu/ handle/2142/11100。[61] M. Wang和C. D. Manning，“用于文本蕴含和问答的结构化潜变量概率树编辑模型”，载于第*23*届国际计算语言学会议（*Coling 2010*）论文集，第1164–1172页，中国北京，2010年8月。[62]S. W.-t. Yih、M.-W. Chang、C. Meek和A. Pastusiak，“使用增强型词汇语义模型进行问答”，载于计算语言学协会第*51*届年会论文集，保加利亚首都索非亚，2013年8月。[63] J. M. Giorgi和G. D. Bader，“迈向生物医学领域可靠的命名实体识别”，《生物信息学》，第36卷，第1期，第280–286页，2020年。

[64] A. Severyn和A. Moschitti，“面向答案选择与抽取的自动特征工程”，载于*2013*年自然语言处理实证方法会议论文集，第458–467页，美国华盛顿州西雅图，2013年10月。[65]S. Wang和J. Jiang，“用于文本序列匹配的比较-聚合模型”，2016年，https://arxiv.org/abs/1611. 01747。[66] S. Yoon、F. Dernoncourt、D. S. Kim、T. Bui和K. Jung，“用于答案选择的潜在聚类比较-聚合模型”，载于第*28*届*ACM*信息与知识管理国际会议论文集，第2093–2096页，中国北京，2019年11月。[67] A. Severyn和A. Moschitti，“利用卷积深度神经网络学习短文本对排序”，载于第*38*届国际*ACM SIGIR*信息检索研究与发展会议论文集，第373–382页，智利圣地亚哥，2015年8月。[68] D. Wang和E. Nyberg，“用于问答中答案句选择的长短期记忆模型”，载于计算语言学协会第*53*届年会暨自然语言处理国际联合会议第*7*届会议论文集，第2卷，第707–712页，中国北京，2015年7月。[69] J. Bromley、J. W. Bentz、L. Bottou等，“使用‘孪生’时延神经网络进行签名验证”，国际模式识别与人工智能期刊，第07卷，第04期，第669–688页，1993年。[70] L. Yu、K. M. Hermann、P. Blunsom和S. Pulman，“答案句选择的深度学习”，2014年，https://arxiv.org/abs/1412.1632。[71]M. Feng、B. Xiang、M. R. Glass、L. Wang和B. Zhou，“将深度学习应用于答案选择：一项研究与一项开放任务”，载于*2015*年*IEEE*自动语音识别与理解研讨会（*ASRU*）论文集，第813–820页，IEEE，美国亚利桑那州斯科茨代尔，2015年12月。[72] H. He、K. Gimpel和J. Lin，“利用卷积神经网络进行多视角句子相似度建模”，载于*2015*年自然语言处理实证方法会议论文集，第1576–1586页，葡萄牙里斯本，2015年9月。[73] J. Rao、H. He和J. Lin，“利用深度神经网络进行答案选择的噪声对比估计”，载于第*25*届*ACM*信息与知识管理国际会议论文集，第1913–1916页，美国印第安纳州印第安纳波利斯，2016年10月。[74]H. T. Madabushi、M. Lee和J. Barnden，“结合问题分类与深度学习以改进答案选择”，载于第*27*届计算语言学国际会议论文集，第3283–3294页，美国新墨西哥州圣菲，2018年8月。[75] S. Kamath、B. Grau和Y. Ma，“将预期答案类型预测并整合到用于答案句选择的简单循环神经网络模型中”，《计算与系统》′，第23卷，第3期，2019年。[76] S. Kumar、K. Dixit和K. Shah，“通过学习词语的上下文敏感影响来解释文本分类器”，载于首届可信自然语言处理研讨会论文集，第55–67页，线上，2021年6月。[77] T. Lai、T. Bui和S. Li，“答案选择中深度学习技术应用综述”，载于第*27*届计算语言学国际会议论文集，第2132–2144页，美国新墨西哥州圣菲，2018年8月。

[78] Y. Zhang和Y. Peng，“基于长短期记忆网络的答案选择研究”，载于*2018*年亚洲语言处理国际会议（*IALP*）论文集，第357–361页，IEEE，印度尼西亚万隆，2018年11月。[79] L. Yang、Q. Ai、J. Guo和W. B. Croft，“aNMM：利用基于注意力的神经匹配模型对短答案文本进行排序”，载于第*25*届*ACM*信息与知识管理国际会议论文集，第287–296页，美国印第安纳州印第安纳波利斯，2016年10月。[80]D. Bahdanau、K. Cho和Y. Bengio，“通过联合学习对齐与翻译实现神经机器翻译”，2014年，https://arxiv.org/abs/1409.0473。[81] G. Shen、Y. Yang和Z.-H. Deng，“用于句对建模的交互加权对齐网络”，载于*2017*年自然语言处理实证方法会议论文集，第1179–1189页，丹麦哥本哈根，2017年9月。[82] Y. Tay、L. A. Tuan和S. C. Hui，“用于文本序列匹配的堆叠残差亲和网络及多层注意力精炼”，2018年，https://arxiv.org/abs/1810.02938。[83] H. He、J. Wieting、K. Gimpel、J. Rao和J. Lin，“马里兰大学-德州大学奥斯汀分校-TTIC在SemEval-2016 任务1中的系统：用于文本相似度测量的基于注意力的多视角卷积神经网络”，载于第*10*届语义评测国际研讨会（*SemEval-2016*）论文集，第1103–1108页，美国加利福尼亚州圣迭戈，2016年6月。[84]L. R. Medsker和L. Jain，“循环神经网络”，设计与应用，第5卷，第64–67页，2001年。[85] S. Abujar、A. K. M. Masum、S. M. H. Chowdhury、M. Hasan和S. A. Hossain，“使用双向循环神经网络进行孟加拉语文本生成”，载于*2019*年第*10*届计算、通信与网络技术国际会议（*ICCCNT*），第1–5页，IEEE，印度坎普尔印度理工学院，2019年7月。[86] A. K. M. Masum、S. Abujar、M. A. I. Talukder、A. S. A. Rabby和S. A. Hossain，“基于序列到序列循环神经网络的抽象式文本摘要方法”，第1–5页，IEEE。[87] Y. Yang、J. Zhou、J. Ai等，“利用对抗长短期记忆网络生成视频描述”，*IEEE*图像处理汇刊，第27卷，第11期，第5600–5611页，2018年。[88]S. Hochreiter和J. Schmidhuber，“长短期记忆”，《神经计算》，第9卷，第8期，第1735–1780页，1997年。[89]A. Graves，“使用循环神经网络生成序列”，2013年，https://arxiv.org/abs/1308.0850。[90]S. V. Moravvej、M. Joodaki、M. J. M. Kahaki和M. S. Sartakhti，“一种基于注意力机制的句子相似度测量方法”，载于*2021*年第7届网络研究国际会议（*ICWR*）论文集，第238–242页，IEEE，伊朗德黑兰，2021年5月。[91] A. Graves和J. Schmidhuber，“利用双向长短期记忆网络及其他神经网络架构进行逐帧音素分类”，《神经网络：国际神经网络学会官方期刊》，第18卷，第5–6期，第602–610页，2005年。[92] J.-´A. Gonz´alez、E. Segarra、F. Garc´ıa-Granada、E. Sanchis和L. r. ı.-F. Hurtado，“用于抽取式摘要的孪生层次注意力网络”，《智能与模糊系统期刊》，第36卷，第5期，第4599–4607页，2019年。[93] C. McCormick，“Word2vec教程——跳元模型”，2016年，
h
t
t
p://
m
c
c
o
r
mick
ml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model。[94] J. Pennington、R. Socher和C. D. Manning，“GloVe：词表示全局向量”，载于*2014*年自然语言处理实证方法会议论文集（*EMNLP*），第1532–1543页，多哈，卡塔尔，2014年10月。[95]S. Thavareesan 和 S. Mahesan，“使用 Word2vec 和 FastText 扩展情感词典，以预测泰米尔语文本的情感”，载于*2020*年莫拉图瓦工程研究会议（*MERCon*）论文集，第272–276页，IEEE，莫拉图瓦，斯里兰卡，2020年7月。[96] H. S. Alatawi、A. M. Alhothali 和 K. M. Moria，“使用领域特定词嵌入、深度学习和 BERT 检测白人至上主义仇恨言论”，*IEEE Access*，第9卷，第106363–106374页，2021年。[97] J. Devlin 和 M.-W. Chang，“开放源代码发布 BERT：自然语言处理领域最先进的预训练”，*GoogleAI Blog*，第2卷，2018年。[98]K. Arulkumaran、M. P. Deisenroth、M. Brundage 和 A. A. Bharath，“深度强化学习：简要综述”，*IEEE Signal Processing Magazine*，第34卷，第6期，第26–38页，2017年。[99] S. A. H. Minoofam、A. Bastanfard 和 M. R. Keyvanpour，“TRCLA：一种减少细胞学习自动机负迁移的迁移学习方法”，*IEEE Transactionson Neural Networks andLearning Systems*，2021年。[100] Z. Movahedi 和 A. Bastanfard，“基于强化学习的马球游戏竞争性多智能体”，*Multimedia Tools and Applications*，第80卷，第17期，第26773–26793页，2021年。[101]M. A. Wiering、H. Van Hasselt、A.-D. Pietersma 和 L. Schomaker，“用于解决分类问题的强化学习算法”，载于*2011*年*IEEE*自适应动态规划与强化学习研讨会（*ADPRL*）论文集，第91–96页，IEEE，巴黎，法国，2011年4月。[102] J. Feng、M. Huang、L. Zhao、Y. Yang 和 X. Zhu，“利用强化学习从噪声数据中进行关系分类”，载于*AAAI*人工智能会议论文集，第32卷，第1页，新奥尔良，路易斯安那州，美国，2018年2月。[103]T. Zhang、M. Huang 和 L. Zhao，“通过强化学习学习用于文本分类的结构化表示”，载于第三十二届*AAAI*人工智能会议论文集，新奥尔良，路易斯安那州，美国，2018年2月。[104] D. Liu 和 T. Jiang，“用于手术手势分割与分类的深度强化学习”，医学图像计算与计算机辅助干预 - *MICCAI 2018*，载于医学图像计算与计算机辅助干预国际会议论文集，第247–255页，施普林格，2018年9月。[105] D. Zhao、Y. Chen 和 L. Lv，“结合视觉注意力的深度强化学习车辆分类”，*IEEETransactions on Cognitive andDevelopmental Systems*，第9卷，第4期，第356–367页，2016年。[106] J. Janisch、T. Pevn´y 和 V. Lis´y，“使用深度强化学习进行高成本特征分类”，*AAAI*人工智能会议论文集，第33卷，第01期，第3959–3966页，2019年。[107] C. Martinez、G. Perrin、E. Ramasso 和 M. Rombaut，“一种用于时间序列早期分类的深度强化学习方法”，载于*2018*年第*26*届欧洲信号处理会议（*EUSIPCO*）论文集，第2030–2034页，IEEE，罗马，意大利，2018年9月。[108] L. Abdi 和 S. Hashemi，“一种针对多类别不平衡数据、基于强化学习的集成剪枝方法”，智能系统进展与

计算，载于第三届软计算问题求解国际会议论文集，第589–600页，施普林格，鲁尔基，印度，2013年12月。[109] A. K. Dixit 和 J. J. Sherrerd，《经济理论中的优化》，牛津大学出版社按需出版，牛津，英格兰，1990年。[110]E. M. Voorhees，“问答系统评估：从TREC问答赛道中汲取的经验”，载于《问答：策略与资源》研讨会项目，第6页，Citeseer，新泽西州，美国，2002年。[111] W. Huang、J. Jiang、Q. Qu 和 M. Yang，“AILA：法律领域的问答系统”，载于第二十九届国际人工智能联合会议论文集，第5258–5260页，横滨，2020年7月。[112] Y. Yang、W.-t. Yih 和 C. Meek，“Wikiqa：开放域问答挑战数据集”，载于*2015*年自然语言处理实证方法会议论文集，第2013–2018页，里斯本，葡萄牙，2015年9月。[113] Y. Shen、Y. Deng、M. Yang 等，“用于问答对排序的知识感知注意力神经网络”，载于第*41*届国际*ACM SIGIR*信息检索研究与发展会议论文集，第901–904页，安娜堡，密歇根州，2018年7月。[114]S. V. Moravvej、M. J. M. Kahaki、M. S. Sartakhti 和 A. Mirzaei，“一种用于问答的双向长短期记忆网络（BLSTM）注意力机制方法”，载于*2021*年第*29*届伊朗电气工程会议（*ICEE*）论文集，第460–464页，IEEE，德黑兰，伊朗伊斯兰共和国，2021年5月。[115] Y. Matsubara、T. Vu 和 A. Moschitti，“用于高效基于Transformer的答案选择的重排序”，载于第*43*届国际*ACM SIGIR*信息检索研究与发展会议论文集，第1577–1580页，中国西安，2020年7月。[116] S. Kim、I. Kang 和 N. Kwak，“基于密集连接循环网络和协同注意力信息的语义句子匹配”，*AAAI*人工智能会议论文集，第33卷，第01期，第6586–6593页，2019年。[117] Y. Song、Q. V. Hu 和 L. He，“P-CNN：e”，《知识工程系统》，第169卷，第67–79页，2019年。[118] G. Bao、Y. Wei、X. Sun 和 H. Zhang，“用于答案选择的双重注意力循环卷积神经网络”，英国皇家学会开放科学，第7卷，第5期，第191517页，2020年。[119] Q. Wang、W. Wu、Y. Qi 和 Z. Xin，“结合主动学习与自定步调学习、利用贝叶斯神经网络进行深度答案选择”，载于*ECAI 2020*论文集，第1587–1594页，IOS出版社，圣地亚哥-德孔波斯特拉，西班牙，2020年7月。[120] W. Huang、Q. Qu 和 M. Yang，“用于答案选择的交互式知识增强注意力网络”，《神经计算与应用》，第32卷，第15期，第11343–11359页，2020年。[121] V. V. Phansalkar 和 P. S. Sastry，“带动量的反向传播算法分析”，《*IEEE*神经网络汇刊》，第5卷，第3期，第505-506页，1994年。[122] M. Hagan、H. Demuth 和 M. Beale，《神经网络设计》，PWS出版公司，波士顿，马萨诸塞州，1996年。[123] C.-C. Yu 和 B.-D. Liu，“一种具有自适应学习率和动量系数的反向传播算法”，载于*2002*年国际神经网络联合会议论文集

网络，第2卷，第1218–1223页，IEEE，美国夏威夷州檀香山，2002年5月。[124] R. Battiti，“学习的一阶和二阶方法：介于最速下降法与牛顿法之间”，《神经计算》，第4卷，第2期，第141–166页，1992年。[125] F. D. Foresee 和 M. T. Hagan，“贝叶斯学习的高斯-牛顿近似”，载于神经网络国际会议论文集（*ICNN’97*），第3卷，第1930–1935页，IEEE，美国德克萨斯州休斯敦，1997年6月。[126] S. Mirjalili、S. M. Mirjalili 和 A. Lewis，“灰狼优化器”，《工程软件进展》，第69卷，第46–61页，2014年。[127] X.-S. Yang，“一种新的蝙蝠启发式元启发式算法”，载于《自然启发式优化协作策略》（*NICSO 2010*）论文集，第65–74页，施普林格，西班牙格拉纳达，2010年5月。[128]S. Mirjalili，“蜻蜓算法：一种用于求解单目标、离散和多目标问题的新型元启发式优化技术”，《神经计算与应用》，第27卷，第4期，第1053–1073页，2016年。[129] D. Bairathi 和 D. Gopalani，“用于训练前馈神经网络的管水母群算法（SSA）”，载于问题求解的软计算，第521–534页，施普林格，新加坡，2019年。[130] X.-S. Yang 和 S. Deb，“基于L´évy飞行的布谷鸟搜索”，载于*2009*年世界自然与生物启发计算大会（*NaBIC*）论文集，第210–214页，IEEE，印度哥印拜陀，2009年12月。[131]S. J. Mousavirad 和 H. Ebrahimpour-Komleh，“人类心智搜索：一种新型基于种群的元启发式优化算法”，《应用智能》，第47卷，第3期，第850–887页，2017年。[132]S. Mirjalili 和 A. Lewis，“鲸鱼优化算法”，《工程软件进展》，第95卷，第51–67页，2016年。[133] D. Karaboga 和 B. Basturk，“一种用于数值函数优化的强大而高效的算法：人工蜂群（ABC）算法”，《全球优化杂志》，第39卷，第3期，第459–471页，2007年。[134] V. Pihur、S. Datta 和 S. Datta，“聚类验证指标的加权秩聚合：一种蒙特卡洛交叉熵方法”，《生物信息学》，第23卷，第13期，第1607–1615页，2007年。[135]S. Xie 和 Z. Tu，“整体嵌套边缘检测”，载于*IEEE*国际计算机视觉会议论文集，第1395–1403页，美国华盛顿特区西北部，2015年12月。[136] T.-Y. Lin、P. Goyal、R. Girshick、K. He 和 P. Doll´ar，“用于密集目标检测的焦点损失”，载于*IEEE*国际计算机视觉会议论文集，第2980–2988页，意大利威尼斯，2017年10月。[137] C. H. Sudre、W. Li、T. Vercauteren、S. Ourselin 和 M. Jorge Cardoso，“广义Dice重叠：用于高度不平衡分割的深度学习损失函数”，载于《医学图像分析中的深度学习与临床决策支持的多模态学习》，第240–248页，施普林格，德国海德堡，2017年。[138] S. S. M. Salehi、D. Erdogmus 和A. Gholipour，“使用3D全卷积深度网络进行图像分割的Tversky损失函数”，《医学成像中的机器学习》，载于国际研讨会论文集

医学成像中的机器学习，第379–387页，加拿大魁北克，施普林格，2017年9月。[139]S. Jadon，“语义分割损失函数综述”，载于*2020*年*IEEE*计算智能生物信息学与计算生物学会议（*CIBCB*）论文集，第1–7页，IEEE，智利维尼亚德尔马，2020年10月。[140] G.Hackeling，精通*Scikit-Learn*机器学习，Packt出版社，英国伯明翰，2014年。[141] S. Sonkar、A. E. Waters 和 R. G. Baraniuk，“注意力词嵌入”，2020年，https://arxiv.org/abs/2006.00988。