

# 量子经典混合算法在抗菌肽设计与 优化方面的应用及创新

第十九届“挑战杯”全国大学生课外学术科技作品竞赛  
“揭榜挂帅”专项赛  
软件设计方案

## 目 录

1 引言与背景介绍 .....	2
1.1 研究意义.....	2
1.2 国内外研究现状.....	2
2 主要结果介绍 .....	4
3 主要方法与原理 .....	5
3.1 数据来源及预处理 .....	5
3.1.1 数据来源 .....	5
3.1.2 数据预处理 .....	6
3.1.3 序列编码 .....	6
3.2 算法原理思路 .....	8
3.2.1 经典部分 .....	8
3.2.2 量子部分 .....	10
1) 编码层	
2) 变分层	
3) 测量层	
4) 极端值的集中处理	
5) 含噪声	
4 结论与展望 .....	16
参考文献 .....	17

# 摘要

作为一种新型的信息处理方法，量子计算基于量子力学揭示的不确定性和非局域性等基本原理进行信息处理，在处理某些问题时，相比经典计算具有指数加速效果。量子深度学习作为量子计算与深度学习的结合，处于前沿研究领域。它充分利用了量子计算机的高效计算能力和深度学习模型的复杂性，探索新的计算范式，在处理高维数据、加速训练过程以及解决传统算法难题方面展示了突出的潜力。其面临大量数据挑战时，量子计算与深度学习的结合有望降低深度学习成本，为深度学习算法的研究带来全新的灵感。

本模型基于机器学习方法实现对于抗菌肽的分析和预测。由于抗菌肽结构复杂，种类繁多，使得其序列呈现出极度分散，长度差异性大，模型训练时，难以学习到其特征部分，效率低下，部分抗菌肽序列结构无特殊性质但被错误学习等问题。针对以上问题，我们使用了 PC6 编码，它提供了肽序列中每个氨基酸的六个理化性质，相较于 PC7、AC6、AC7 等编码方法表现出更高的精确度和特异性，且在外部测试数据上的表现与内部测试数据相当，具有良好的泛化能力，高效的信息传递能力。同时我们剔除极小部分极度不规则序列，并进行序列数据截断以最大限度保留其特征信息，达到更好的预测分析效果。

本文构造了量子长短期记忆神经网络(Quantum Long Short-Term Memory, QLSTM)，用于抗菌肽理化性质的预测。混合量子-经典长短期记忆神经网络主要由变分量子线路 (Variational Quantum Circuit, VQC) 代替传统神经网络，门控机制通过量子门操作量子比特的叠加态和纠缠态，使其控制信息的流动方式更加复杂且多样，且同时保留了长期依赖记忆的能力，旨在利用量子计算的特性来进一步优化记忆和信息处理的效率。通过数值实验，在大样本中噪声的抗菌肽理化性质预测任务中，模型预测理化性质与其真实理化性质数值高度吻合，在精确度，学习效率上均优于传统长短期记忆神经网络。

# 1 引言与背景介绍

## 1.1 研究意义

据《柳叶刀》的研究，2019 年超过 120 万人死于抗生素耐药性感染，高于艾滋病毒（HIV）/艾滋病（AIDS）或疟疾造成的死亡<sup>[1]</sup>，而抗生素的市场需求仍不断上升，因此急需一种可代替抗生素发挥抗菌作用的药物。天然基因编码的抗菌肽（AMPs）是一类在自然界广泛存在的先天免疫分子，通常含有 12 至 100 个氨基酸残基。它们在大多数生命形式中均有发现，如细菌素、真菌肽抗生素、植物硫蛋白和昆虫、两栖动物及高等脊椎动物的防御素<sup>[1-3]</sup>。AMPs 主要作为第一线的防御分子，通过破坏微生物的细胞膜完整性来抑制或杀死病原体<sup>[4]</sup>，其作用机制通常是通过在细胞膜上形成孔隙导致渗透休克<sup>[5]</sup>。与传统抗生素相比，AMPs 不针对特定的靶点，而是针对微生物细胞的普遍物理化学属性，可通过多种不同的机制来抑制细菌生长，因此较少引发耐药性问题<sup>[6,7]</sup>，因此，在抗生素耐药性日益严重的今天，AMPs 被视为治疗感染的有希望的替代方案<sup>[8]</sup>。

在生物医学研究领域，特别是在 AMPs 的理化性质预测方面，科研人员面对的主要挑战之一是如何处理随技术进步急剧增加的海量数据。传统计算方法在处理这些高维度、大规模的数据集时往往受到计算资源的限制，难以满足对高效率及高精度的严格需求。量子计算以其理论上的超线性速度优势，提供了一种突破传统计算限制的新途径。特别是在 AMPs 蛋白质序列分析及其理化性质预测中，量子算法通过量子位的叠加和纠缠等独特性质，实现了指数级的计算速度提升，从而极大地增强了数据处理的效率和结果的精确性。在这种背景下，我们开发的量子经典混合算法引入了量子长短时记忆网络（QLSTM）和变分量子线路（VQC）。这种算法结构利用了量子计算在处理复杂模型时的高效性，同时融合了经典算法的稳定性和可靠性。通过 VQC，我们实现了从纯经典算法到量子经典混合算法的平滑过渡，有效地利用量子线路的参数化特性来优化算法的学习和预测能力。这不仅为抗菌肽的理化性质提供了更加精确的预测模型，也显著缩短了分析时间，为生物活性分子的功能评估与优化提供了强大的科学支持。

随技术进步与需求的增长，特别是在个性化医疗和精准健康管理领域的扩展中，对于如抗菌肽等生物大分子的深入研究提出了更高的计算需求。量子与经典计算的混合算法的研发，融合了量子计算的高速处理能力与经典计算的广泛适用性，为精确预测抗菌肽的理化性质提供了一个强有力的新工具。基于量子计算的算法在预测抗菌肽的理化性质方面的开发，有望能满足医药研发及健康产业对高效算力的迫切需求。这种研究不仅与技术发展趋势相符，而且具有广泛的应用前景，预计将在生物医药领域产生深远影响。通过促进抗菌肽及其他生物活性分子的应用与开发，有望为未来的医疗治疗提供更多可能性。

## 1.2 国内外研究现状

2007 年，Sneh Lata 等人提出了基于机器学习方法分析和预测抗菌肽<sup>[9]</sup>。近些年，更深层次的机器学习模型用于多肽的预测，如人工神经网络（ANN），定量矩阵（QM）和支持向量机（SVM）等，都是有效的计算方法，可以加快小分子药物的发现<sup>[10-12]</sup>。因此，基于机器学习的方法作为一种耗时更少、成本效益更高的方法，在识别新的潜在肽方面得到了广泛关注<sup>[13]</sup>。尤其是抗菌肽活性的预测方面，已经采用复杂的方法来提高这些肽作为候选药物的效力，或基于先前的知识和分子模拟发现设计具有抗菌活性的新序列<sup>[14]</sup>。机器学习，特别是深度学习，可以从复杂的数据集中提取有用的信息<sup>[15]</sup>。例如，AmPEP 预测模型<sup>[16]</sup>是利用氨基酸分布模式编码的一组 AMP 序列训练的随机森林来实现的；iAMPpred 模型<sup>[17]</sup>基于支持向量机（SVM），编码时考虑了残基的组成和理化性质以及结构性质；Veltri 等人<sup>[18]</sup>通过神经网络嵌入层对 AMP 进行编码，然后利用深度学习方法开发 AMP 分类器。

“量子深度学习”概念源发于“量子计算”与“深度学习”两大系统性理论的交叉融合。得益于量子运算强大的并行运算处理能力与深度学习模型的自我优化性能。量子深度学习在生命科学领域的应用开发近年来已然创造出了阶段性成果：在生物信息学方面，量子深度学习可赋能实现生物数据聚类任务、赋能完成基因表达数据与宏观生物类群的拟态任务<sup>[19]</sup>。借助量子深度学习的赋能作用，相关研究人员能够更为充分地理解生物学数据的内在性关联与系统性规律<sup>[20]</sup>；在基因组学方面，通过量子深度学习的赋能，基因测序片段的模拟过程可被大幅优化，进而充分提高基因组组装的完成效率与拟合性。例如，在量旋科技以及华大生命科学研究所在的合作过程中，研发团队充分利用了量子算法来解决基因组组装过程中的结构性技术壁垒<sup>[21]</sup>；医药研发方面，在量子深度学习的技术加持下，研究人员可更加准确地预测药物分子与靶细胞的结合过程与亲和性能，进而更加充分地实现药物活性预测功能；此外，量子深度学习可赋能“生物医药学数据运算”过程，量子深度学习能够在极大程度上加速分子数据库的数据提取筛选过程，进而实现分子并行筛选功能；在蛋白质结构预测方面，得益于量子深度学习的赋能作用，国内外科研团队能够快速高效地完成“蛋白质折叠过程中分子相互作用力与能量变化过程”的模拟与识别，进而更加准确地预测蛋白质的生物学结构与生理学特性<sup>[22]</sup>。

量子深度学习技术的引入不仅为传统生命科学领域开辟了更多研究发展方向，极大程度上地优化了该领域研究模态的底层逻辑、加速了实验数据聚类环节的整合效率。由此可见，量子深度学习为抗菌肽的预测提供了新的可能性。

表 1. 量子算法在生物学和医学研究中的应用概览

序号	目标应用	硬件设备	算法类型	经典复杂性	预期优势	备注
1	蛋白质折叠与构象模拟	量子退火器	量子退火	多项式；启发式近似	最高达多项式级别	适用于生物物理和量子生物化学模拟
2	分子对接模拟	高斯玻色子采样器	抽样	超多项式	最高达超多项式级别	分子生物学和量子生物化学的应用
3	序列比对	通用门控量子设备	优化	多项式；启发式近似	多项式	应用于基因组学和遗传学
4	序列匹配	通用门控量子设备	量子机器学习、搜索	多项式	最高达超多项式级别	包括 DNA 分析和病原体识别
5	系统发生树推断	通用门控量子设备	优化	超多项式	多项式	应用于生物多样性研究
6	生物网络推断	量子退火器	优化	多项式和超多项式	多项式	适用于细胞和遗传网络分析
7	转录因子结合分析	量子退火器	优化	多项式；启发式近似	未知，最高达多项式级别	应用于调控基因表达研究
8	神经网络	通用门控量子设备	量子机器学习	多项式和超多项式	多项式，问题特定，测量变化	量子神经网络用于复杂数据处理
9	de novo 组装	量子退火器，通用门控量子设备	量子退火、优化	多项式；启发式近似	未知，最高达多项式级别	重建未知或复杂 DNA 序列

## 2 主要结果介绍

在本赛中，我们选择了量子经典混合算法的应用这一方向进行研究；选择抗菌肽的序列和十二项性质作为数据集后，我们在最开始使用了经典的长短期记忆网络（LSTM）算法进行了一定的测试，得到不错的结果之后，将其转为量子计算，即得到 QLSTM，由于无噪声环境过于理想化，所以我们选择在有噪声的情况下进行测试，测试后得到不错的结果，在训练过程中预测结果与实际结果逐渐拟合。与经典的 LSTM 算法相比，量子算法速度更快，精度也更高。

表 2. LSTM 与 QLSTM 算法在第 1, 3, 6, 11 轮训练结果的比较（X 轴：Sample index；Y 轴：Target value）

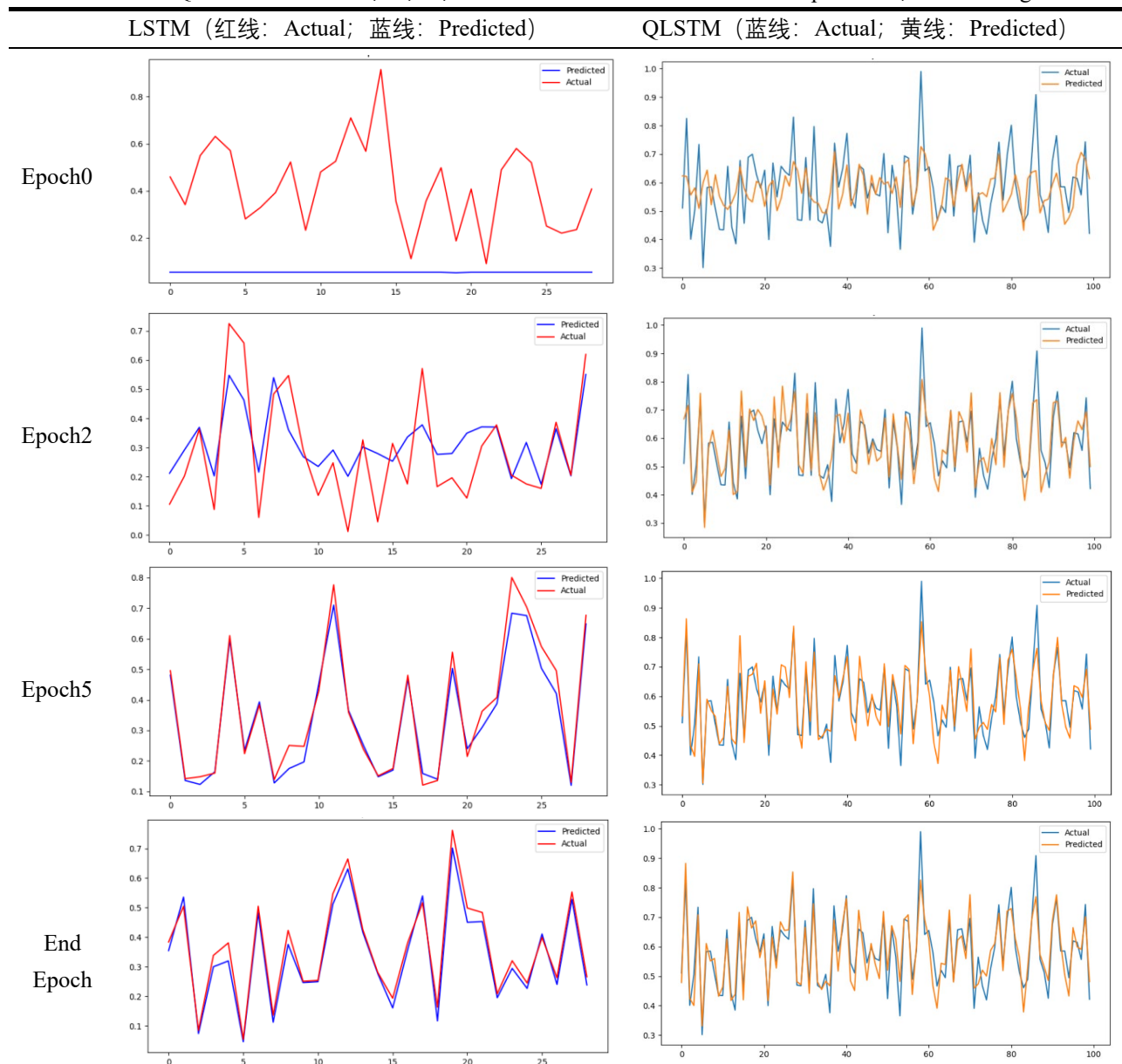


表 3. 两种算法的初始与最终训练 loss 值的比较

	初始 loss	结果 loss
LSTM	0.31405	0.00018
QLSTM	0.24000	0.00037

### 3 主要方法与原理

#### 3.1 数据来源及预处理

##### 3.1.1 数据来源

数据库随着抗菌肽数量变化而急剧增加, 如何处理大量分散的数据是一项具有挑战性的任务。总的来说, 在大量文献中, 许多肽及其相关信息如序列、活性等是分散的<sup>[12]</sup>。基于多肽的生物信息学资源可以促进多肽的管理、搜索和挖掘, 进一步加快多肽药物预测和设计模型的建立。因此, 有必要选取高质量的基于肽的数据库数据作为模型的数据来源。如表 1 所示, 总结了目前权威的基于抗菌肽的数据库。

在 2015 年, Pirtskhalava 等人建立了一个名为“Database of Antimicrobial Activity and Structure of Peptides”(DBAASP)<sup>[13]</sup>的数据库。该数据库汇集了来自 PubMed 的超过 8000 条记录, 并根据合成途径的不同, 将其分类为三种主要类型: 核糖体合成的多肽超过 2000 条, 非核糖体合成的多肽约 80 条, 以及合成多肽超过 5700 条。DBAASP 专注于多肽的化学结构及其通过实验测定的最小抑制浓度(MIC)、半数抑制浓度(IC50)等抗菌活性参数。此外, 该数据库还提供了多肽的溶血活性和细胞毒性数据。

进一步地, 在 2016 年, Fan 等人开发了一个名为“Database of Antimicrobial Peptides”(DRAMP)<sup>[14]</sup>的数据库, 旨在全面收集和保存抗菌肽的信息。DRAMP 包含了抗菌肽的多种注释, 如蛋白质数据银行(PDB)条目、MIC 值、专利信息、临床数据、结构数据和参考文献。该数据库收录了 17,349 个抗菌肽序列, 其中包括 12,704 个专利序列, 4,571 个通用抗菌肽序列, 以及 74 个特定肽序列, 并提供了多种序列分析工具, 例如序列相似性搜索、序列比对和保守域搜索。

相比于其他数据库, DBAASP 数据库提供了更详细的抗菌肽理化性质信息, 因此我们选择 DBAASP 数据库作为训练数据来源, 并与 DRAMP 数据库进行验证确保数据的正确性。

表 4. 抗菌肽的主要数据库

数据集	年份	网址	肽类型	参考文献
APD	2003	<a href="http://aps.unmc.edu/AP/main.html">http://aps.unmc.edu/AP/main.html</a>	综合	[15, 16]
AMPer	2007	<a href="http://www.cnbi2.com/cgi-bin/amp.pl">http://www.cnbi2.com/cgi-bin/amp.pl</a>	前体和成熟体	[17]
PhytAMP	2009	<a href="http://phytamp.pfba-lab-tun.org">http://phytamp.pfba-lab-tun.org</a>	植物	[18]
CAMP	2009	<a href="http://www.camp.bicnirrh.res.in/">http://www.camp.bicnirrh.res.in/</a>	成熟	[19, 20]
DAMPD	2011	<a href="http://apps.sanbi.ac.za/dampd">http://apps.sanbi.ac.za/dampd</a>	前体和成熟体	[21]
YADAMP	2012	<a href="http://www.yadamp.unisa.it">http://www.yadamp.unisa.it</a>	生物资源	[22]
AVPdb	2013	<a href="http://crdd.osdd.net/servers/avpdb">http://crdd.osdd.net/servers/avpdb</a>	抗病毒	[23]
DBAASP	2015	<a href="http://dbaasp.org/">http://dbaasp.org/</a>	核糖体和非核糖体	[13]
DRAMP	2016	<a href="http://dramp.cpu-bioinfor.org/">http://dramp.cpu-bioinfor.org/</a>	通用、临床和专利	[14]

### 3.1.2 数据预处理

由于数据量较大，我们采用了牛顿插值法对数据进行插值，以填补数据库中的缺失值。在分析过程中，鉴于数据集中标签值的数量较多，我们选择了归一化疏水矩、归一化疏水性、净消耗、等电点、穿透深度、倾角、无序构象倾向、线性力矩、体外聚集的倾向、疏水残基的影响角度、两亲性指数、线圈的倾向十二个相关的属性进行后续计算。经过逐步回归的分析，我们发现体外聚集的倾向性质对回归结果有显著影响。进一步的深入研究表明，该性质与肽链的空间结构密切相关，仅通过序列信息不能进行有效预测。因此，我们决定在预测模型中排除该性质，而将其预测问题留待后续研究解决。

为了提升神经网络在预测极端值和小数据量情况下的效果，我们采用了箱型图来剔除离群数据。剩余数据的集中性在可接受的范围内，有利于神经网络的后续训练。在数据中发现了重复现象后，我们观察到一些肽链结构长度不同但组成相同，导致它们的相关性质完全相似。由于这种情况下的数据处理具有挑战性，且长短期记忆网络（LSTM）具有遗忘机制，因此我们并未完全清除这些数据，而是选择性地去除了影响较大的极端数据组。

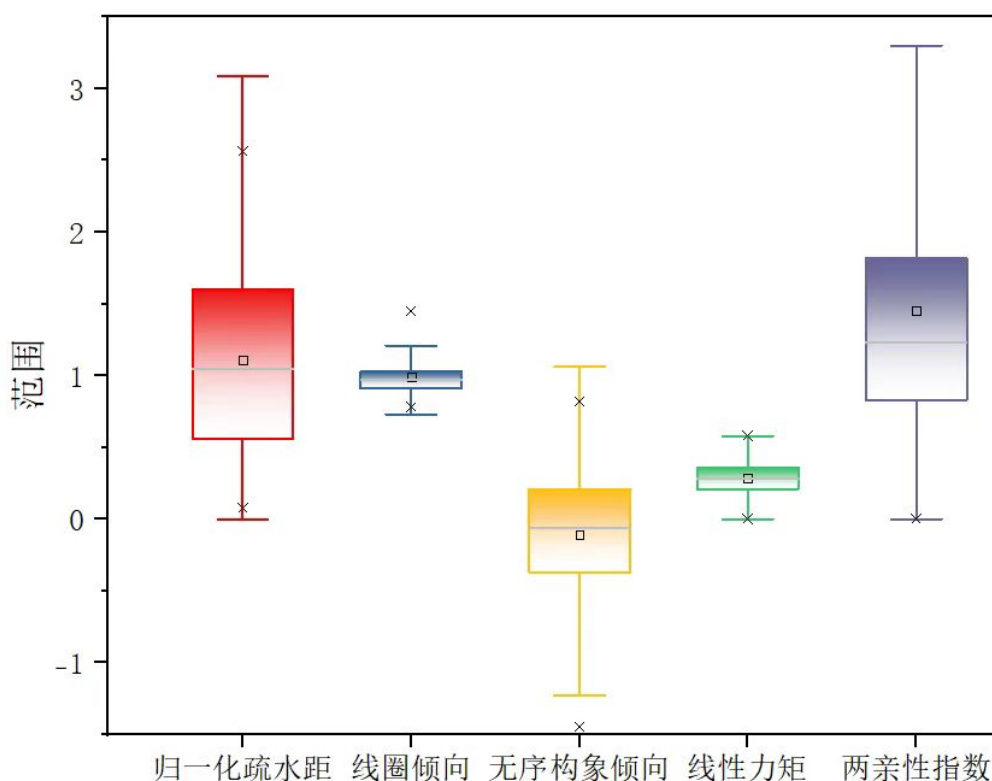


图 1. 单一分布随机数据的箱形图比较

### 3.1.3 序列编码

PC6 是一种新的蛋白质编码方法，它提供了肽序列中每个氨基酸的六个理化性质。在使用外部测试数据时，PC6 相较于 PC7、AC6、AC7 等编码方法表现出更高的精确度和特异性，且在外部测试数据上的表现与内部测试数据相当，具有良好的泛化能力<sup>[24]</sup>。与传统的 one-hot 编码方式相比，PC6 在位置编码的基础上可以进行性质编码，具有更好的相关性，并且显著减少了模型的输入维度，有助于降低计算资源的消耗。因此，我们选择使用 PC6 编码方式将预处理后的抗菌肽序列转化为矩阵形式，以便输入深度学习模型<sup>[25]</sup>。

PC6 编码的构建过程基于 R 包<sup>[26]</sup>中提供的 115 种理化性质数据。首先,采用层次聚类方法对这 115 个性质进行分析,并根据内部平方和的最小化原则确定了六个最优的簇数量。从每个簇中选择一个代表性的理化性质,最终确定了疏水度 (H1)、侧链体积 (V)、极性 (P1)、等电点 pH 值 (pI)、-COOH 基团的解离常数 (pKa) 和侧链净电荷指数 (NCI) 这六个性质的 PC6 编码的基础。接下来,通过 amino\_encode\_table\_6 函数生成了包含以上六个理化性质的编码表,并对每个属性的数值范围进行了标准化,以确保各属性在编码中的权重均衡和一致性。

为了处理不同长度的序列,我们引入了填充字符 0,并编写了 padding\_seq 函数,以确保所有序列具有统一的长度。然而,填充后的长序列可能导致神经网络训练过程中的梯度消失问题,甚至使得网络预测结果完全相同。为了解决这些问题,我们进行了长度的升序排列,并根据图 1 中显示的序列长度分布图进行了分析。我们发现长度超过 41 的序列占比不到总数的 10%。因此,我们决定剔除这些较长的序列,并对剩余的数据进行进一步的处理和筛选,确保每个输入样本都不含填充字符 0。

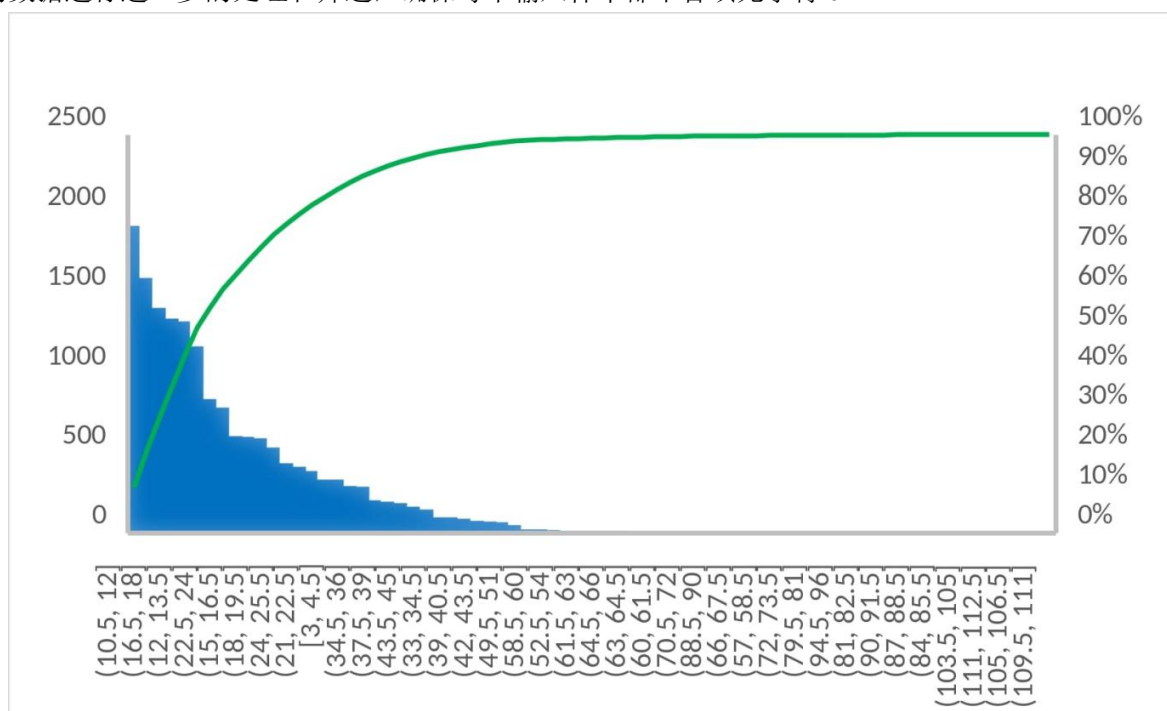


图 2. 序列长度排列图

PC\_encoding 函数则利用 PC6 编码表对填充后的序列进行编码,将每个抗菌肽序列转换为一个 40 元素的向量,并将每个元素替换为相应的 PC6 编码表中的六维向量,从而生成一个  $40 \times 6$  的矩阵。最终,编码后的矩阵被保存为 JSON 文件,作为用于训练深度学习模型的最终数据集。

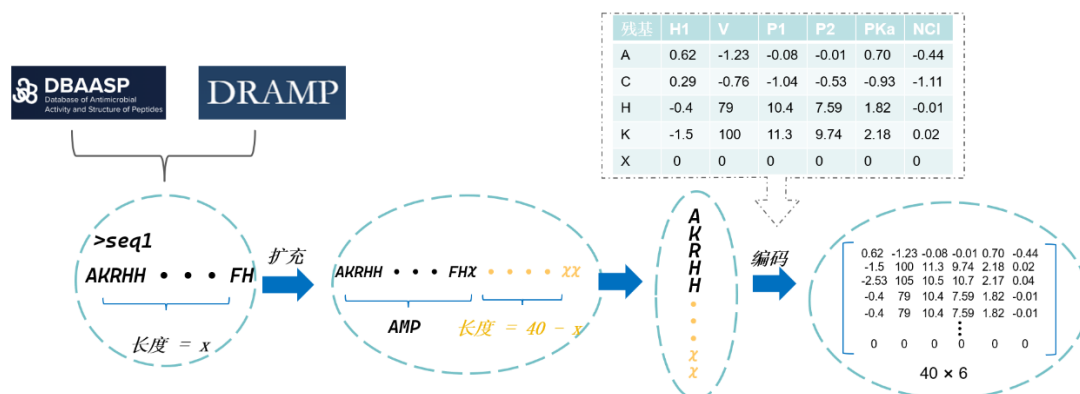


图 3. PC6 编码过程图



## 3.2 算法原理思路

### 3.2.1 经典部分

在设计经典神经网络的部分模型中, 采用了长短时记忆网络 (LSTM) 作为核心结构, 专门处理抗菌肽序列的 PC6 编码特征。LSTM 属于循环神经网络 (RNN) 的一种, 具备学习和处理长期依赖信息的能力, 因此非常适合序列数据的分析。循环神经网络 (RNN) 是专门处理序列数据的神经网络, 通过网络内部的循环连接来表达时间或序列结构。在 RNN 中, 信息沿序列在网络内传递, 使得网络能够利用之前的信息来影响当前输出。RNN 的核心特征在于其隐藏层, 这些层不仅连接到后续层, 还回连到自身, 形成内部循环, 保持信息流的持续性, 实现对序列动态变化的记忆。然而, 标准 RNN 模型易受梯度消失或爆炸的影响, 限制了其学习长距离依赖的能力。为克服这一限制, LSTM 作为 RNN 的改进版本被提出。通过引入几个门控机制 (输入门、遗忘门和输出门), LSTM 可以在不同程度上控制信息的存储或遗忘, 从而有效维持长期依赖, 极大提升了模型处理长序列数据的能力。在本项目中, LSTM 网络的配置如下:

- 1) 输入层大小 (input\_size): 控制输入特征的维度, 即每个输入向量的大小, 本项目中设为 6。
- 2) 隐藏层大小 (hidden\_size): 定义在 LSTM 层中每个单元或神经元的输出特征数量, 本项目中为 64。
- 3) 输出层大小 (output\_size): 网络输出的特征数量, 本项目为预测任务设定为 1。
- 4) 层数 (num\_layers): LSTM 网络的层数, 本项目中使用单层结构。

LSTM 的数据流和层级架构可以通过以下公式表示:

- 1) 输入门:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (1)$$

其中,  $i_t$  是输入门的输出。 $\sigma$  是 sigmoid 激活函数, 用于将值压缩到 0 到 1 之间, 表示信息的接收程度。 $W_i$  是输入门的权重。 $b_i$  是输入门的偏置。 $[h_{t-1}, x_t]$  是前一个隐藏状态与当前输入的连接。输入门的作用是控制当前输入  $x_t$  和前一个时间步的隐藏状态  $h_{t-1}$  的信息量, 决定多少新信息被允许加入到单元状态中。

- 2) 遗忘门:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2)$$

其中,  $f_t$  是遗忘门的输出, 指示保留旧信息的比例。 $W_f$ 、 $b_f$  分别是遗忘门的权重和偏置。遗忘门的作用是决定保留多少之前的单元状态  $C_{t-1}$ 。

- 3) 输出门:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (3)$$

其中,  $o_t$  是输出门的输出, 控制信息的输出量。 $W_o$ 、 $b_o$  分别是输出门的权重和偏置。输出门的作用是决定当前单元状态  $C_t$  中的哪部分信息将输出为当前的隐藏状态  $h_t$ 。

- 4) 单位更新:

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (4)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (5)$$

其中,  $\tilde{C}_t$  是新的候选值。 $*$  表示逐元素乘积。 $C_{t-1}$  是前一时间步的单元状态。 $i_t$ 、 $f_t$  控制信息的流入和遗忘。单元状态是 LSTM 的核心, 用于在整个序列中传递相关信息。其结合了过去的状态和新的候选值来更新其值。

- 5) 隐藏状态:

$$h_t = o_t * \tanh(C_t) \quad (6)$$

其中,  $\sigma$  表示 sigmoid 激活函数,  $*$  表示 Hadamard 积 (元素间乘积),  $W$  和  $b$  代表权重和偏置参数。最终的输出是通过连接最后的隐藏状态  $h_t$  到一个全连接层 (fc), 并转化为最终预测结果。

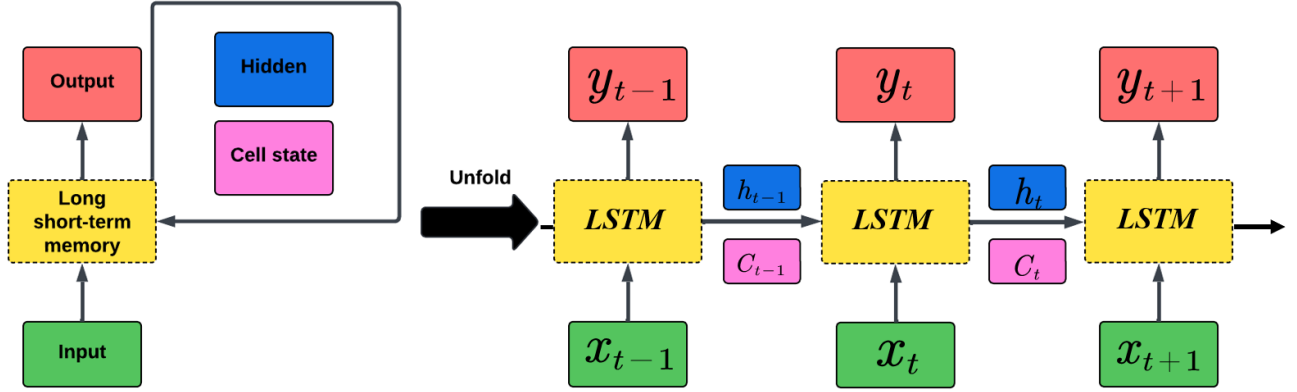


图 4. LSTM 网络结构图

图 i. 展示了 LSTM 单元的操作流程。在实际应用中, 模型首先将输入数据通过 LSTM 层处理, 再通过全连接层 (Linear layer) 转换为最终的输出预测, 我们仅采用序列的最后一个输出。通过上述配置和代码实现, 我们可以有效地处理经过 PC6 特征编码的抗菌肽序列, 并预测其理化性质。这种模型不仅在抗菌肽研究中具有应用价值, 同时也展示了深度学习在序列数据处理方面的强大能力。

模型训练步骤是深度学习工作流程的关键组成部分, 最初使用经典的优化算法替换像 Adam 一样的高级优化器以调整网络参数以最小化损失函数。下面显示了具体的训练步骤及其相应的公式:

1) 初始化和配置:

首先初始化模型和优化器, 并设置学习速率(lr)和将模型传送至计算设备。使用均方误差 (Mean Squared Error, MSE) 作为损失函数, 这是评估模型预测与实际值差异的常用方法。

2) 批处理和迭代:

带有 DataLoader 的数据集, 以批处理的方式和随机打乱的方式将数据传递到模型的训练过程中。每个训练周期 (epoch) 中, 模型通过数据加载器接收输入  $x$  和目标  $y$ , 执行前向传播操作以获取预测  $y\_pred$ 。

3) 损失计算与反向传播:

损失函数的值计算如下:

$$Loss = \frac{1}{n} \sum_{i=1}^n y_i - y\_pred_i \quad (7)$$

在模型训练过程中, 损失函数的选择对模型的性能有直接影响。均方误差 (MSE) 作为损失函数, 该表达式计算了预测值和实际值之间差值的平方和的平均值。Adam 优化器通过自适应地调整各参数的学习率, 有效地最小化了这一损失函数。其中,  $y_i$  是实际值,  $y\_pred_i$  是模型预测值。通过调用 `loss.backward()` 函数自动计算模型参数相对于损失的梯度。

4) 优化步骤:

使用 `optimizer.step()` 函数更新模型的权重, 以最小化损失值。在每次迭代后, 清除旧的梯度信息以准备下一批数据的处理。

5) 性能评估:

记录并输出每个训练周期后的损失值, 以监控训练进度。通过绘制损失变化曲线, 可以直观地看到模型在学习过程中的性能变化。损失曲线通常应显示向下的趋势, 表明模型在逐渐学习数据的特征并改进其

预测。

在训练结束后，通过绘制损失曲线，可以清晰地看到模型在学习过程中的性能变化。损失曲线通常应显示向下的趋势，表明模型在逐渐学习数据特征并改进其预测。通过上述描述和代码实现，可以确保经典神经网络部分的模型训练是按照科学的方法进行，以期达到高精度的预测结果。

### 3.2.2 量子部分

量子 LSTM (QLSTM)模型是经典 LSTM 模型的量子版本。与经典 LSTM 模型相比，QLSTM 使用 VQC 代替了不同门的经典神经网络。通过用 VQC 代替神经网络，QLSTM 学习更快，收敛更稳定<sup>[26]</sup>。

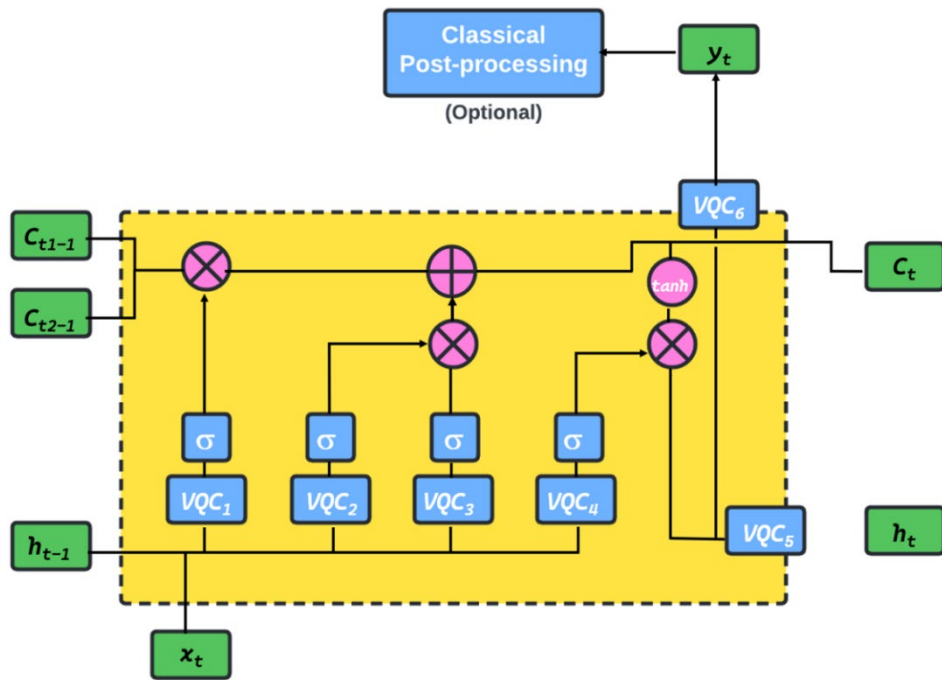


图 5. QLSTM 结构

变量子电路 (VQC) 是一种具有可调参数且为迭代优化而设计的量子电路。VQC 的基本思想是利用可变量子电路将输入数据编码到量子态上。通过量子门和测量操作，对这些量子态进行操作，提取与分类任务相关的特征。随后，处理测量结果，并将其用于为输入数据分配类别标签。VQC 将经典优化技术与量子计算相结合。在训练过程中，可变量子电路在量子计算机或模拟器上重复执行，将结果与训练数据的真实标签进行比较。通过迭代调整可变量子电路的参数，优化过程旨在最小化代价函数，即预测标签与真实标签之间的差异。该优化过程旨在找到最优的量子电路配置，从而最大化预测准确性。

先前结果表明，这种电路对量子噪声具有鲁棒性，因此适合于 NISQ 器件。VQC 已成功应用于函数逼近<sup>[27]</sup>、分类<sup>[28,29]</sup>、生成建模<sup>[30]</sup>、深度强化学习<sup>[31]</sup>和迁移学习<sup>[32]</sup>。此外，有研究指出，VQC 比经典神经网络更具表现力<sup>[33,34]</sup>，因此可能比后者更好。在这里，表达能力是指用有限数量的参数表示某些函数或分布的能力。事实上，人工神经网络(ANN)被认为是通用逼近器<sup>[35]</sup>，这意味着一个神经网络即使只有一个隐藏层，理论上也可以近似任何可计算函数。正如我们将在下面看到的，使用 VQC 作为量子 LSTM 的构建块可以实现更快的学习。

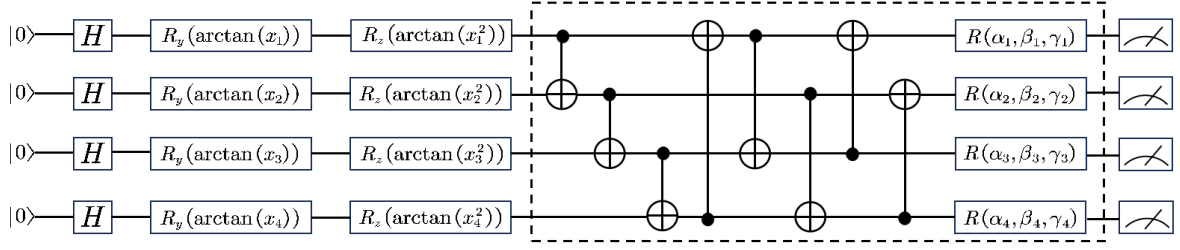


图 6. 用于 QLSTM 的 VQC 架构

量子 LSTM (QLSTM)模型是经典 LSTM 模型的量子版本。与经典 LSTM 模型相比, QLSTM 使用 VQC 代替了不同门的经典神经网络。通过用 VQC 代替神经网络, QLSTM 学习更快, 收敛更稳定<sup>[36]</sup>。结构如图 2 所示, 对应的正向传递方程以如下形式表示为:

$$f_t = \sigma(VQC_1(v_t)) \quad (8)$$

$$i_t = \sigma(VQC_2(v_t)) \quad (9)$$

$$\tilde{C}_t = \tanh(VQC_3(v_t)) \quad (10)$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{C}_t \quad (11)$$

$$o_t = \sigma(VQC_4(v_t)) \quad (12)$$

$$h_t = VQC_5(o_t * \tanh(c_t)) \quad (13)$$

$$y_t = VQC_6(o_t * \tanh(c_t)) \quad (14)$$

### 1) 编码层

经过编码处理的矩阵的每个向量对应抗菌肽序列中的每个氨基酸, 向量将与细胞该时间部的状态进行拼接, 之后经过线性层变换成与量子比特对应大小的向量, 输入进 QLSTM 神经网络。上述数据在经过整合模拟后将会进入编码层, 将经典数据值映射为量子振幅。然后 Hadamard 门将对输入的四个量子比特进行系统化处理加工, 最终形成无偏态数据, 以便进行角度旋转。我们使用 Angle Embedding 进行角度编码: 作为一种基于量子力学原理的量子化编码方式, 角度编码能够利用量子比特相位的对应性来实现信息数据的量化存储与实化传输。具体而言, 在量子计算体系中存在两组对应关系: 复数向量与量子比特一一对应、角度与量子相位一一对应, 因此, 信息数据的编码解码过程能够通过量子比特相位的泛化调整过程来实现。

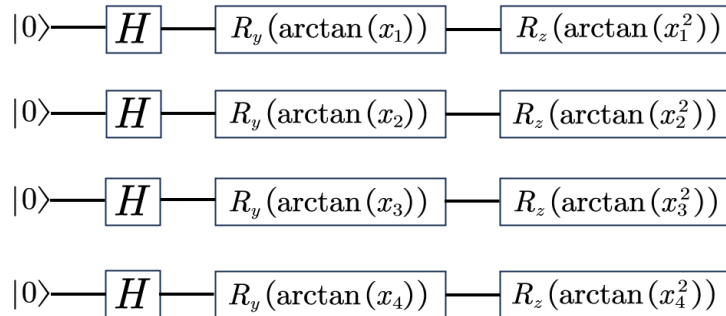


图 7. 编码层

在量子态准备过程中，我们利用量子旋转门将量子比特初始化为特定的量子态，通过利用量子计算的并行性和量子纠缠特性，确保量子比特在初始化后的状态能够充分确定为叠加态或纠缠态。在 Angle Embedding 函数中，每个经典输入值将被先后转换为  $R_y$  门和  $R_z$  门的旋转角度，对输入的量子比特分别进行  $\arctan(x)$  和  $\arctan(x^2)$  处理。这样，每个量子比特就根据输入数据被旋转到了一个特定的角度，从而实现了输入信息的编码。通过这种方式，有效地建立了经典数据与量子系统之间的接口，改变了量子比特的量子态。

在量子算法执行阶段，一旦量子比特通过数据预处理、量子态准备和角度编码步骤正确实现初始化编码，我们可以充分利用量子比特的叠加态、纠缠性和干涉性，加速将经典数据转化为量子比特的技术步骤。

## 2) 变分层

作为参数化的量子电路，Ansatz 是对编码器电路从原始数据中产生的特征进行处理，并输出新的特征用于接下来的回归任务。可调参数是在神经网络算法中确定的，因此它们可根据样本数据进行优化。使用 Ansatz 来逼近目标函数，可将特征数据映射到不同预测值。相似于与经典神经网络中的泛函近似定，总存在一种量子线路，可以在任意小误差范围内逼近目标函数<sup>[37]</sup>，此外也可用多项式量子门构建电路<sup>[38]</sup>，因此通过优化变分量子线路也可实现模型的优化。在实践中，变分线路是由量子门搭建的，设计线路有两种方法：受问题启发方法和硬件高效方法。受问题启发的线路结构，实现方法是构建问题的哈密顿量，并将其至于线路中，运行线路得到测量结果，这样的线路结构有利于收敛。

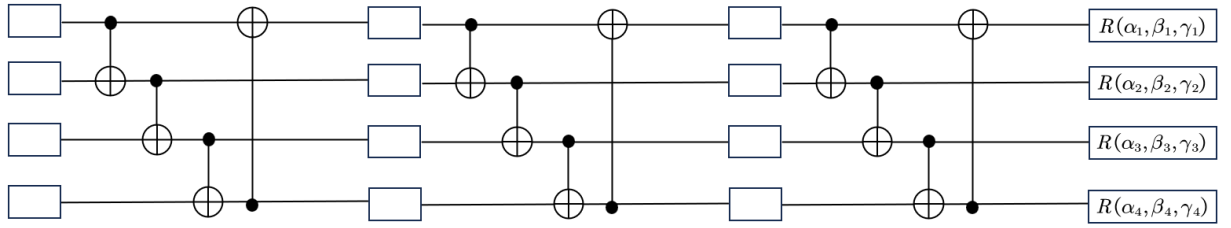


图 8. 变分层

2018 年由谷歌提出的随机量子线路 RQC 为典型的硬件高效线路结构，核心思想是通过多项式个随机选择的两比特门组成的量子线路即可产生充分的纠缠，两比特门多选择受控量子旋转门和 CNOT 门，通过量子比特间的纠缠，使其输出态逼近整个希尔伯特空间。QLSTM 中使用的变分线路通常是硬件高效电路，由特定的双量子比特门和单量子比特门组<sup>[39]</sup>，本质上这种分析可以根据特定量子器件的门集和连通性进行定制，并且可以直接在器件上实现而无需编译。

抗菌肽序列编码后的数据处于量子态，然后将经历一系列的单一化操作。这些量子操作如图 1 由数个 CNOT 门和单量子比特旋转门组成。此处调用 pennylane 封装的 BasicEntanglerLayers 函数，其会布置一个单参数单量子旋转的层，CNOT 门以循环的方式应用于每一对具有固定邻接 1 和 2 的量子比特衔接在旋转层后，以产生多量子比特纠缠。作为变分量子电路的关键特性，CNOT 门的组合可以使量子比特之间产生复杂的纠缠关系。由于有四个量子，这里的纠缠层层数为四。需要注意的是虽然有参数 weight 将量子态旋转，但是 weight 并不参与训练，仅负责在每次纠缠后将量子旋转一部分角度。随后接入旋转门 R，该门包含三个参数  $\alpha$ ， $\beta$ ， $\gamma$ 。但此处旋转门中沿 x、y、z 轴的 3 个旋转角并非先前固定，而是在基于梯度下降法的迭代优化过程中进行更新。

## 3) 测量层

在变分量子电路 (VQC) 的测量层，我们将对量子态进行测量，以提取与分类任务相关的特征。通过这些测量操作，我们能够从量子比特中获得经典数据，并将其用于进一步的处理和分析。

在量子态经过编码层和变分层的处理后，我们对每个量子比特进行测量，以提取与分类任务相关的特

征。具体而言，我们采用 Pauli-Z 基进行测量，Pauli-Z 测量是量子计算中的标准测量基，其结果将直接反映量子比特在 Z 轴上的投影，为后续数据处理提供必要的经典信息，通过计算每个量子比特的期望值（expectation value），我们可以获取这些量子比特的经典信息。在测量操作过程中，我们通过将量子态投影到经典状态来完成从量子信息到经典数据的转换。通过将处理过的量子态在 Pauli-Z 基上进行多次重复测量来计算每个量子比特的期望值，我们能够获取稳定且可靠的经典数据，减少量子噪声对测量结果的影响。

测量结果是经典数据，通过多次重复测量以统计方式处理这些数据，可以显著减少由于量子噪声引入的误差。统计分析通过重复实验和计算平均值来平滑噪声干扰，从而提升数据的可靠性和准确性。这些处理后的测量结果形成特征向量的一部分，这些特征向量被进一步输入到经典机器学习模型中，用于执行分类或回归任务。

量子态测量层通过将量子信息转换为经典数据，为后续的机器学习任务提供了关键的输入。通过多次测量和统计处理，我们能够有效减少噪声对测量结果的影响，确保测量数据的可靠性和准确性。结合编码层和变分层的处理，测量层在量子噪声环境下依然能够提供稳定可靠的数据支持，确保算法的整体性能和稳定性。

#### 4) 极端值的集中训练处理

在训练我们的量子经典混合算法时，我们注意到某些极端值对模型学习效果产生了显著影响。由于这些极端值在预测中表现不佳，我们决定对这些数据进行集中训练，以提高模型的整体性能和对极端值的处理能力。这一过程涉及对数据进行筛选、集中特训以及结果整合等多个步骤。

首先，我们在数据处理中剔除了对模型训练产生负面影响的错误极端值。这些极端值通常由于数据采集过程中的噪声或异常情况导致，剔除这些数据有助于提高模型的整体训练质量。然而，另一部分极端值并非由于错误或噪声引起，而是数据本身的自然变异。这些极端值可能包含重要的生物信息或特性，对这些数据的有效处理对于提升模型的实际应用能力至关重要。因此，我们对这些合理的极端值进行了集中训练处理。

在筛选过程中，我们根据数据分布的上下百分位数来确定哪些数据属于极端值。将筛选出的极端值数据从整个训练集中分离出来，形成一个独立的极端值训练集。对这些极端值数据进行多次单独训练，通过调整模型参数和优化策略，增强模型对极端值的适应性。采用较小的学习率和更精细的参数调整方法，确保模型能够有效学习到极端值的特征。

在完成极端值的集中训练后，我们将优化后的模型参数与原始模型进行融合，形成最终的混合模型。通过对比集中特训模型与原始模型的预测性能，我们选择最优的融合方法，将两者的优势结合起来。这一过程通常包括加权平均或其他模型融合技术。

为了验证最终模型的性能，我们使用了验证集和测试集，特别关注其对极端值的预测能力。通过交叉验证和多次实验，我们确保模型的稳定性和泛化能力。实验结果表明，经过集中训练处理的极端值数据，模型在面对这些数据时表现出了更高的预测准确性和鲁棒性。

这种处理方法不仅合理且有效，通过剔除噪声和错误极端值，保证了模型训练的纯净度；通过集中训练合理的极端值，提升了模型对重要生物信息和特性的捕捉能力。最终，使得我们的量子经典混合算法能够在实际应用中展现出更高的可靠性和稳定性。这种全面的极端值处理策略确保了模型在复杂生物数据环境下的卓越表现，为进一步的生物信息分析和应用提供了坚实的基础。

#### 5) 含噪声测试

本次算法设计中，我们将充分考虑到量子计算机在实际运行中经常面临的噪声干扰问题，并专门集成了先进的、专门处理量子噪声的算法模块。该模块通过引入和模拟各种噪声模型，有效提高了算法在噪声环境下的鲁棒性和坚固性。它的实现提高了量子算法在噪声干扰下的性能，并进一步显著提升了实际应用中的可靠性和稳定性，对量子计算的实现具有决定性的意义。

为了有效模拟现实世界量子计算机的环境，我们在算法中引入了多种噪声测试，以评估和优化算法的



性能和鲁棒性:

首先,我们在算法中定义了一些用于添加噪声的条件和噪声操作:

1) 条件定义

- c0: 操作是 ‘qml. PauliX’ 或 ‘qml. PauliY’
- c1: 操作是 ‘qml. Hadamard’ 且操作的量子位是 [0, 1]
- c2: 操作是 ‘qml. RX’
- c3: 操作是 ‘qml. RY’ 且参数大于等于 0.5

2) 噪声操作

- n0: 对操作应用振幅阻尼噪声, 参数为 0.4
- n1: 对操作应用热弛豫噪声, 指定了多个时间常数和参数
- n2: 修改 ‘RX’ 操作的参数, 缩放因子为 0.05
- n3: 对操作应用相位阻尼噪声, 参数为 0.9

3) 噪声模型

noise\_model: 将条件 c0, c1, c2 与对应的噪声操作 n0, n1, n2 绑定, 并增加条件 c3 与 n3 的绑定, 构成一个完整的噪声模型。

**振幅阻尼噪声(n0):** 此噪声模拟了量子比特由于与环境相互作用从高能态向低能态弛豫的过程, 我们可以观察到量子比特在能量损失情况下的表现, 进而使我们能够评估系统在实际应用中的能量衰减情况下的稳定性和计算准确性。具体来说, 我们使用 ‘qml.noise.partial\_wires’ 方法定义了振幅阻尼噪声模型, 如下所示: ‘n0 = qml.noise.partial\_wires(qml.AmplitudeDamping, 0.4)’, 其中 ‘qml.AmplitudeDamping’ 指定了振幅阻尼噪声类型, 参数 ‘0.4’ 表示噪声强度。

振幅阻尼信道的表达形式为:

$$\varepsilon_{AD}(\rho) = E_0 \rho E_0^\dagger + E_1 \rho E_1^\dagger \quad (15)$$

其中  $E_0, E_1$  是 Kraus 算符,  $\gamma$  为耗散系数。

$$E_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix}, E_1 = \begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix} \quad (16)$$

**热弛豫噪声(n1):** 热弛豫噪声通过模拟量子比特在特定环境温度下的弛豫过程, 涉及 T1 和 T2 时间常数。热弛豫噪声的引入让我们能够评估量子比特在不同温度条件下的稳定性和可靠性, 有助于我们观察和理解噪声对量子态保持时间的影响, 并优化我们的模型以在特定温度条件下更好地运行。T1 时间常数描述了量子比特从激发态到基态的能量弛豫, 而 T2 时间常数则描述了量子比特相干性的损失。通过引入热弛豫噪声, 我们可以评估量子比特在不同温度条件下的稳定性和可靠性, 并分析噪声对量子态保持时间的影响。具体来说, 我们使用 ‘ThermalRelaxationError’ 方法来定义热弛豫噪声模型, 如下所示:

‘ThermalRelaxationError(0.4, metadata[“t1”], 0.2, 0.6, op.wires)’。参数 ‘0.4’ 代表噪声强度, 而 ‘metadata[“t1”]’ 其他参数则指定了 T1 和 T2 时间常数, 以及噪声的具体特性。

热弛豫信道的表达形式为:

$$\varepsilon_{TR}(\rho) = \sum E_i \rho E_i^\dagger \quad (17)$$

其中, 具体的 Kraus 算子  $E_i$  取决于 T1 和 T2 时间常数以及环境温度。

**参数缩放噪声(n2):** 此噪声通过对量子门操作参数进行缩放来模拟控制误差, 这种噪声类型帮助我们理解由于控制不精确引入的参数误差如何影响整体计算精度。通过调整和优化这些参数, 我们能够显著提高量子计算的准确性, 从而提升我们算法在处理抗菌肽序列时的预测性能。具体来说, 我们使用以下代码定义了参数缩放噪声模型: ‘qml.RX(op.parameters[0] \* 0.05, op.wires)’。在这段代码中, ‘qml.RX’

表示对量子比特执行旋转操作，‘`op.parameters[0] * 0.05`’是对原始参数进行缩放，模拟控制误差。

参数缩放信道的表达形式为：

$$\varepsilon_{PS}(\rho) = U(\theta + \Delta\theta) \rho U^\dagger(\theta + \Delta\theta) \quad (18)$$

其中， $U(\theta)$ 表示一个量子门操作，其参数 $\theta$ 会受到误差 $\Delta\theta$ 的影响，导致参数缩放。

**相位阻尼噪声(n3):** 此噪声类型模拟了量子位的相位信息丢失，常见于相干性损失。通过相位阻尼噪声测试，我们可以分析量子系统在相干性受到干扰时的表现，评估量子系统的相干时间和抗干扰能力。具体来说，我们使用‘`qml.noise.partial_wires`’方法定义了相位阻尼噪声模型，如下所示：

‘`qml.noise.partial_wires(qml.PhaseDamping, 0.9)`’，‘`qml.PhaseDamping`’表示相位阻尼噪声类型，参数‘0.9’表示噪声强度。这一设置模拟了量子比特相位信息的损失，使我们能够评估和分析这种噪声对量子计算的影响。

相位阻尼信道的表达形式为：

$$\varepsilon_{PD}(\rho) = E_0 \rho E_0^\dagger + E_1 \rho E_1^\dagger \quad (19)$$

其中， $\gamma$ 为损失系数。

$$E_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix}, E_1 = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{\gamma} \end{bmatrix} \quad (20)$$

通过这些噪声测试的分析结果，我们发现我们的算法在面对不同类型的噪声时表现出了良好的鲁棒性。具体表现为：

表 5. 量子算法抗噪性能分析

噪声类型	影响	抗噪表现
振幅阻尼噪声(n0)	造成能量损失	算法依然能够保持较高的计算精度，证明了其对能量衰减的耐受性
热弛豫噪声(n1)	不同温度和弛豫时间常数	算法展示出较为良好的稳定性和可靠性，表明其在各种环境条件下都能有效运行
参数缩放噪声(n2)	存在控制误差	算法通过适当调整参数依然能够实现高精度计算，显示了其在参数误差情况下的自适应能力
相位阻尼噪声(n3)	相干性丢失	算法依然能够保持相对稳定的性能，证明其在抗干扰方面具有较强的能力



## 4 结论与展望

我们的基础算法 LSTM 算法是一种特殊的循环神经网络算法，通过门控机制丢弃或保留所需信息，然后通过反向传播的方式使门的参数更好的适应数据，在此基础之上，我们通过量子电路 VQC 代替神经网络，将经典优化技术与量子计算相结合，找到最优配置，由此得到我们需要的 QLSTM 模型。

我们选择利用抗菌肽序列对其 12 种性质进行预测，首先在经典 LSTM 算法中，能够取得不错的成效，预测有一定的准确度，但数据量庞大，会浪费很多资源；所以我们选择使用 QLSTM 算法，在经过拥有不同噪声环境的测试下，虽然资源和时间限制，但量子算法在少资源情况下得到的预测结果依旧可观；未来我们的模型有望应用于从序列到性质再到结构的分子理性设计当中，作为抗菌肽生成式 AI 模型 hydramp 的限制性生成算法，或对已有的抗菌肽性质与功能进行优化。我们的工作是在量子与经典混合算法中的重要尝试与实践，在大规模信息预测与生物信息方面提供了一定的参考价值。

本质上，我们提出的模型是经典 LSTM 算法的量子化模型，是对循环神经网络的优化与改进；与传统的算法相比，LSTM 算法在自然语言处理、时间序列预测和语音识别等领域有着广阔的应用前景，在这里我们用到的是对自然语言处理，能够取得很好的效果；同时由于使用了 VQC 代替了不同门的经典神经网络，我们的算法可以找到最优的量子电路配置，从而学习得更快，最大化预测准确性；由于算法的量子化，叠加和纠缠原理使存储容量和计算效率得到了增强，在图像上显示为 loss 收敛的更快，也允许了并行处理和改进的数据表示；通过量子特性来维护和处理更长的序列，对解决经典 LSTM 的梯度消失问题有很大的作用。

如今的生物科学研究中常常涉及到巨量的数据，如基因组数据、蛋白质组数据等，并且数据量的快速增长是其发展趋势。量子计算机在处理这些大数据时，能够利用其超高的并行性和量子态叠加原理，提供比传统计算更快的数据处理能力。因此，量子计算有望高效的模拟计算量极大的模型，例如大分子相互作用和复杂的代谢网络，这对于传统计算机来说非常困难的。由此可以加快药物发现和设计的周期，推动个性化医疗和精准医疗的实现。但是如何节约量子资源，如何让我们的模型有更高的实用性和有效性，都是接下来需要进一步研究的方向。目前来说，这些工作可以从数据处理，模型改进以及量子运算三个方面进行改进，像是将 12 种性质进行统一处理来一次性预测，尝试利用两种计算范式的优势，使数据处理更高效、更有效，设计更合理的量子线路来节约资源和提高性能，以及对于极端值和数据较少的部分进行额外的训练或采取更合适的模型等方式；这些对我们的模型都有针对性的改进。

如今，量子计算已经在众多科学领域发挥了至关重要的作用，现在的很多研究也让我们看到在处理复杂生物信息学等生命科学问题上其巨大的潜力；随着量子计算机技术的不断发展和成熟，我们算法的速度和精度将得到进一步的提高，当前我们利用的 LSTM 算法适用范围很广，有很大的发展空间，同时与其他的像是强化学、生成对抗网络等技术相结合，其应用场景将会进一步扩大，也将取得更显著的成果。

## 参考文献

- [1] THE LANCET: An estimated 1.2 million people died in 2019 from antibiotic-resistant bacterial infections, more deaths than HIV/AIDS or malaria. EurekaAlert. Retrieved 19-JAN-2022.
- [2] J. B. McPhee and R. E. Hancock, "Function and therapeutic potential of host defence peptides," *J Pept Sci*, vol. 11, pp. 677–687, 2005.
- [3] M. R. Yeaman and N. Y. Yount, "Unifying themes in host defence effector polypeptides," *Nat Rev Microbiol*, vol. 5, pp. 727–740, 2007.
- [4] K. A. Brogden, "Antimicrobial peptides: pore formers or metabolic inhibitors in bacteria?" *Nat Rev Microbiol*, vol. 3, pp. 238–250, 2005.
- [5] M. Magana, M. Pushpanathan, A. L. Santos, L. Leanse, M. Fernandez, A. Ioannidis, M. A. Giulianotti, Y. Apidianakis, S. Bradfute, A. L. Ferguson, A. Cherkasov, M. N. Seleem, C. Pinilla, C. de la Fuente-Nunez, T. Lazaridis, T. Dai, H. Houghten.
- [6] B. Aslam, W. Wang, M. I. Arshad, M. Khurshid, S. Muzammil, M. H. Rasool, M. A. Nisar, R. F. Alvi, M. A. Aslam, M. U. Qamar, M. K. F. Salamat, and Z. Baloch, "Antibiotic resistance: a rundown of a global crisis," *Infect Drug Resist*, vol. 11, pp. 1645–1658, 2018.
- [7] S. J. Baker, D. J. Payne, R. Rappuoli, and E. De Gregorio, "Technologies to address antimicrobial resistance," *Proc Natl Acad Sci U S A*, vol. 115, pp. 12887–12895, 2018.
- [8] R. Spohn, L. Daruka, V. Lázár, A. Martins, F. Vidovics, G. Grézal, O. Méhi, B. Kintsés, M. Számel, P. K. Jangir, B. Csörgö, Á. Györkei, Z. Bódi, A. Faragó, L. Bodai, I. Földesi, D. Kata, G. Maróti, B. Pap, R. Wirth, B. Papp, and C. Pál, "Integrated evolutionary analysis reveals antimicrobial peptides with limited resistance," *Nat Commun*, vol. 10, p. 4538, 2019.
- [9] Sneha Lata; BK Sharma; GPS Raghava. Analysis and prediction of antibacterial peptides. *BMC Bioinformatics* 2007, 8:263 <http://doi.org/10.1186/1471-2105-8-263>
- [10] Fang, J.; Yang, R.; Gao, L.; Yang, S.; Pang, X.; Li, C.; He, Y.; Liu, A.L.; Du, G.H. Consensus models for CDK5 inhibitors in silico and their application to inhibitor discovery. *Mol. Divers.*, 2015, 19(1), 149-162. <http://dx.doi.org/10.1007/s11030-014-9561-3>
- [11] Fang, J.; Li, Y.; Liu, R.; Pang, X.; Li, C.; Yang, R.; He, Y.; Lian, W.; Liu, A.L.; Du, G.H. Discovery of multitarget-directed ligands against Alzheimer's disease through systematic prediction of chemical-protein interactions. *J. Chem. Inf. Model.*, 2015, 55(1), 149-164. <http://dx.doi.org/10.1021/ci500574n>
- [12] Fang, J.; Yang, R.; Gao, L.; Zhou, D.; Yang, S.; Liu, A.L.; Du, G.H. Predictions of BuChE inhibitors using support vector machine and naive Bayesian classification techniques in drug discovery. *J. Chem. Inf. Model.*, 2013, 53(11), 3009-3020. <http://dx.doi.org/10.1021/ci400331p>
- [13] Shah, Y.; Sehgal, D.; Valadi, J.K. Recent trends in antimicrobial peptide prediction using machine learning techniques. *Bioinformatics*, 2017, 33(12), 415-416. <http://dx.doi.org/10.1093/bioinformatics/btx135>
- [14] Maccari G, Di Luca M, Nifosì R. 2015. Computational peptidology, p195–219. Springer, New York, NY.
- [15] LeCun Y, Bengio Y, Hinton G. 2015. Deep learning. *Nature* 521:436–444. <https://doi.org/10.1038/nature14539>.
- [16] Bhadra P, Yan J, Li J, Fong S, Siu SWI. 2018. AmPEP: sequence-based prediction of antimicrobial peptides using distribution patterns of amino acid properties and random forest. *Sci Rep* 8:1697.
- [17] Meher PK, Sahu TK, Saini V, Rao AR. 2017. Predicting antimicrobial peptides with improved accuracy by incorporating the compositional, physico-chemical and structural features into Chou's general PseAAC. *Sci Rep* 7:42362. <https://doi.org/10.1038/srep42362>.

- 
- [18] Veltri D, Kamath U, Shehu A. 2018. Deep learning improves antimicrobial peptide recognition. *Bioinformatics* 34:2740–2747.
- [19] G. E. Hinton and R. R. Salakhutdinov, “Reducing the Dimensionality of Data with Neural Networks,” *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006
- [20] N. Wiebe, A. Kapoor, and K. M. Svore, “Quantum Deep Learning,” May 2015
- [21] S. Garg and G. Ramakrishnan, “Advances in Quantum Deep Learning: An Overview,” May 2020.
- [22] I. Cong, S. Choi, and M. D. Lukin, “Quantum Convolutional Neural Networks,” *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, Dec. 2019
- [23] Qureshi, A.; Thakur, N.; Tandon, H.; Kumar, M. AVPdb: a data base of experimentally validated antiviral peptides targeting medically important viruses. *Nucleic Acids Res.*, 2014, 42(Database is sue), D1147-D1153.
- [24] Tzu-Tang Lin,<sup>a</sup> Li-Yen Yang,<sup>a</sup> I-Hsuan Lu,<sup>a</sup> Wen-Chih Cheng,<sup>a</sup> Zhe-Ren Hsu,<sup>a</sup> Shu-Hwa Chen,<sup>b</sup> Chung-Yen Lina: AI4AMP: an Antimicrobial Peptide Predictor Using Physicochemical Property-Based Encoding Method and Deep Learning
- [25] Porto W, Fensterseifer I, Ribeiro S, Franco O. 2018. Joker: an algorithm to insert patterns into sequences for designing antimicrobial peptides. *Biochim Biophys Acta Gen Subj* 1862:2043–2052.
- [26] Chen SYC, Yoo S, Fang YLL (2022) Quantum long short-term memory. In: ICASSP 2022-2022 IEEE Int. Conf. Acoust. Speech Signal Process. ICASSP. IEEE, pp 8622–8626
- [27] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, “Quantum circuit learning,” *Physical Review A*, vol. 98, no. 3, p. 032309, 2018.
- [28] M. Schuld, A. Bocharov, K. Svore, and N. Wiebe, “Circuit-centric quantum classifiers,” *arXiv preprint arXiv:1804.00633*, 2018.
- [29] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, “Supervised learning with quantum-enhanced feature spaces,” *Nature*, vol. 567, no. 7747, pp. 209–212, 2019
- [30] P.-L. Dallaire-Demers and N. Killoran, “Quantum generative adversarial networks,” *Physical Review A*, vol. 98, no. 1, p. 012324, 2018.
- [31] S. Y.-C. Chen, C.-H. H. Yang, J. Qi, P.-Y. Chen, X. Ma, and H.-S. Goan, “Variational quantum circuits for deep reinforcement learning,” *IEEE Access*, vol. 8, pp. 141007–141024, 2020
- [32] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, “Transfer learning in hybrid classical-quantum neural networks,” *arXiv preprint arXiv:1912.08278*, 2019.
- [33] T. Lanting, A. J. Przybysz, A. Y. Smirnov, F. M. Spedalieri, M. H. Amin, A. J. Berkley, R. Harris, F. Altomare, S. Boixo, P. Bunyk, et al., “Entanglement in a quantum annealing processor,” *Physical Review X*, vol. 4, no. 2, p. 021041, 2014.
- [34] Y. Du, M.-H. Hsieh, T. Liu, and D. Tao, “The expressive power of parameterized quantum circuits,” *arXiv preprint arXiv:1810.11922*, 2018.
- [35] K. Hornik, M. Stinchcombe, H. White, et al., “Multilayer feedforward networks are universal approximators,” *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.
- [36] Chen SYC, Yoo S, Fang YLL (2022) Quantum long short-term memory. In: ICASSP 2022-2022 IEEE Int. Conf. Acoust. Speech Signal Process. ICASSP. IEEE, pp 8622–8626
- [37] Cai, H., Ye, Q., & Deng, D.-L. (2022). Sample complexity of learning parametric quantum circuits. *Quantum Science and Technology*, 7(2), Article 025014.
- [38] Benedetti, M., Lloyd, E., Sack, S., & Fiorentini, M. (2019). Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4), Article 043001.
- [39] Kandala, A., Mezzacapo, A., Temme, K., Takita, M., Brink, M., Chow, J. M., et al. (2017). Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671), 242–246.