学校代码： 10246
学　　号：12210240046

硕 士 学 位 论 文

（科学学位）

基于主题模型和深度学习的元基因组序列归类

Metagenomic Sequences Binning Based on Topic Model and Deep learning

院系：　　计算机科学技术学院

专业：　　计算机软件与理论

姓名：　　张瑞昌

指导教师：　　周水庚 教授

完 成 日 期：　　2015 年 4 月 10 日

指导小组成员名单

周水庚　教授

目录

第一章 引言

1.1 研究背景及意义

在生物学中，一个生物体的基因组是指包含在该生物的DNA中的全部遗传信息，又称基因体。基因组包括基因和非编码DNA，更精确地讲，一个生物体的基因组是指染色体中的完整DNA序列。

生命科学及研究技术的迅速发展，使得人们对生命现象的了解越来越深入。越来越多的研究者关注微生物，因为它在工业、农业、医疗卫生、环境保护等各方面的重要地位。自然状态下，微生物几乎无处不在，无论是在自然环境如土壤、海洋甚至一些极端环境中，还是在人类和动物的皮肤、肠道中，微生物都与它们所在的环境相伴相生。

除生存环境极为广泛以外，微生物的数量还极为庞大，以人类为例，人类的基因总数只占人类身上微生物基因总数的1%[1]左右。这些微生物是环境能量、物质代谢的重要中间环节和组成部分，它们有些可以代谢生成周围其他生物所必需的底物，而有些则会代谢生成毒性物质，导致环境污染，或者宿主的疾病。因此，对微生物的研究显得极为重要。微生物的传统研究方法主要是依赖将微生物进行培养和分离（culture-dependent）。然而，到目前为止，绝大多数微生物（99% 以上）不能依靠这样的方式获得，这极大地限制了人们对微生物的研究。随着测序技术和数据处理分析能力的飞速发展，以及人们对微生物之间相互依存的共生互利和平衡关系的深入认识，一种可以对环境中所有微生物进行研究而不依赖培养的新方向——宏基因组学应运而生。

以人类基因组计划为代表的生物体基因组研究成为整个生命科学研究的前沿，而微生物基因组研究又是其中重要的分支。世界权威性杂志<<科学>>曾将微生物基因组研究评为世界重大科学进展之一。

由于生物实验的局限性，传统的微生物基因组学经常关注于单个人的细菌基因组。然而，环境中的微生物基因组通常会互相产生影响。例如，人类中的微生物已被证明与常见疾病有关如炎性肠病(IBD)[2]和胃肠紊乱[3]。

1.1.1 元基因组

如图1.1所示，宏基因组学(环境基因组学或生态基因组学)是直接从环境样本，例如人体肠道，土壤，空气中的尘埃中直接研究遗传物的学科。随着新一代测序(NGS)技术的快速发展，我们可以直接对混合环境的DNA样品获得的多个物种进行测序。宏基因组序列来自多个微生物基因组，并且通常大多数宏基因组序列所在的物种名是未知的。

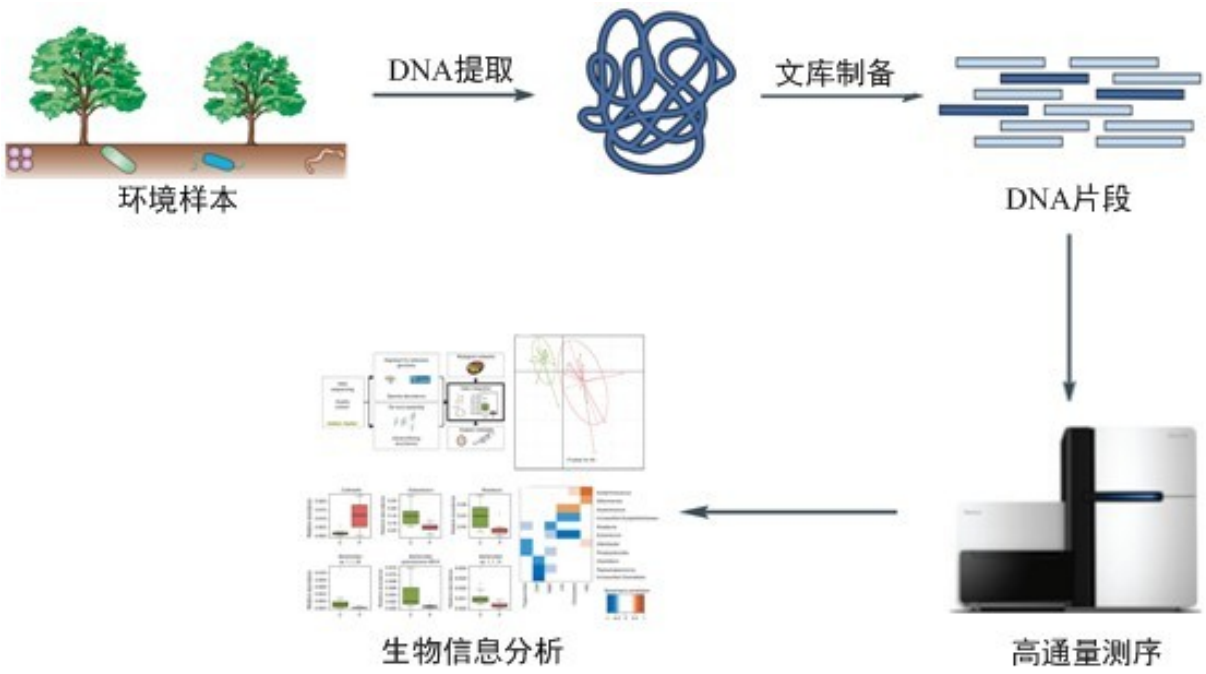

图 1.1: 宏基因组测序分析技术路线[4]。

1.1.2 元基因组归类

与传统的基因组测序研究不同的是，宏基因组研究一个环境中所有物种的DNA序列，且测序得到的DNA序列来自不同的物种。宏基因组归类[5]是指将包含多个物种的DNA片段进行分离，并将每个物种的DNA 序列划分成一个类，如图1.2所示。 由于下一代测序技术产生的DNA序列比较短，少于几百bp，这也给宏基因组序列归类问题带来了新的挑战。同时多数DNA序列的物种名是未知的，如何对DNA 序列进行有效聚类和分析也是至关重要的问题[6]。

1.2 本文内容

本文提出了基于主题模型和基于深度学习的元基因组归类方法。针对本文的特定问题，本文主要工作可以分为以下几个方面：

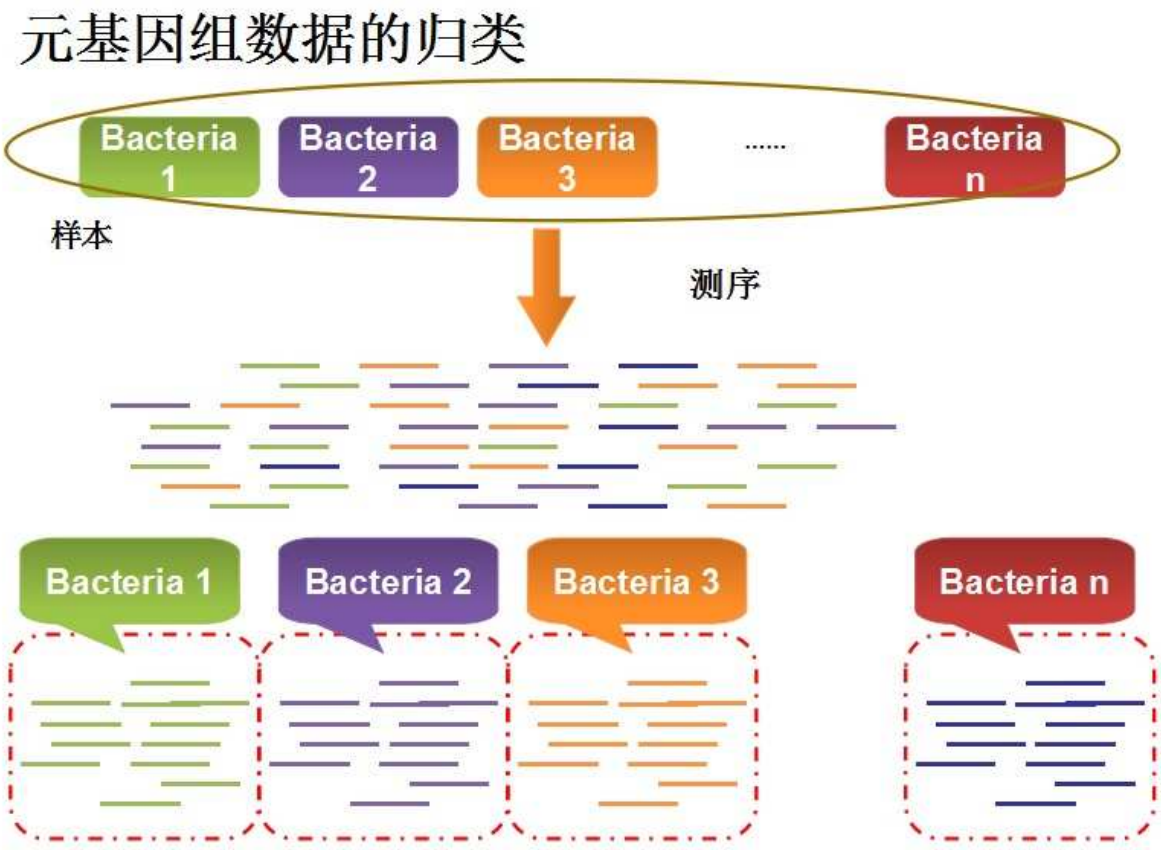

图 1.2: 元基因组采样及归类概述。

1. 提出了基于主题模型的元基因组聚类算法—TM-MCluster，该算法首先从序列中提取k-mer特征，然后采用主题模型对特征向量进行空间变换，最后用SKWIC算法对序列进行聚类。

2. 采用深度学习中的自动编码机对k-mer特征进行特征学习，然后用SKWIC算法对序列进行聚类

3. 构建了涵盖长序列(平均长度1000bp)，短序列(平均长度128bp), 高丰度(相对测序深度为10),大数据集(100W条序列)等各种条件的模拟数据集。

4. 提出了三个聚类算法的评价标准，并对各个算法进行全方位的性能分析。

5. 测试了LDA建模过程中，主题个数对于聚类效果的影响。

6. 分析了多个聚类算法的时间和空间消耗。

7. 我们的方法在长序列上和MetaCluster 3.0、AbundanceBin以及MCluster算法进行了比较分析，对于短序列与MetaCluster 5.0 进行了比较分析。

1.3 本文结构

本文的组织结构如下：

第二章中，我们将介绍元基因组归类问题的相关工作。具体而言，该章详细介绍了目前已有的聚类算法，包括基于序列相似度和序列组成成分分析两类方法，随后介绍了主题模型及其在许多生物问题中的应用。

第三章中，将详细介绍TM-MCluster方法的详细流程。首先从序列中抽取k-mer特征，然后将LDA运用到元基因组序列中，使基于k-mer的词频向量转化为主题向量，然后运用SKWIC算法进行聚类。

第四章中，将详细介绍实验部分。首先详细介绍模拟数据集和真实数据集的参数，以及TM-MCluster算法与其他聚类算法在各个数据集上的实验结果，并将他们进行了全方面的比较分析。

第五章中，将详细介绍深度学习的概念，以及自动编码机。首先从序列中抽取k-mer特征，然后用自动编码机对k-mer特征进行特征学习，然后采用SKWIC算法进行聚类。

第六章中，对全文进行总结，基于主题模型的算法能够有效提高元基因组序列的聚类效果，基于深度学习的算法也取得了不错的聚类效果，并提出未来的工作方向。

第二章 基于主题模型的元基因组聚类算法

本文提出的方法包含三步：1) 用k-mer频率向量来表示序列 2) 通过LDA模型，将每个k-mer频率向量转化为主题分布向量 3) 和MCluster一样，用SKWIC算法对向量化的序列进行聚类。图 2.1中显示了TM-MCluster的工作流程，随后将给出每一步的具体细节。

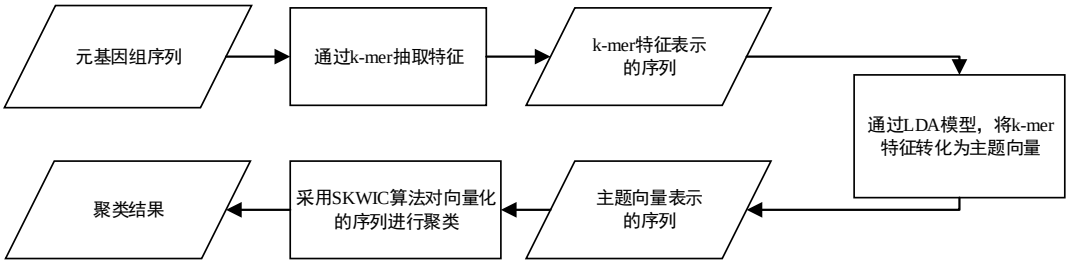

图 2.1: TM-MCluster的工作流程。

2.1 基于k-mer的元基因组序列特征提取

一般来说，k-mer表示序列中k个连续字符组成的子序列。元基因组数据来自不同物种的很多序列组成，我们使用k-mer来刻画序列的特征。DNA序列一共有4 种不同的核苷酸，因此一个DNA序列中最多有4_k个k-mer。一个序列对应的k-mer频率作为其中的一个分量。为了减少计算量，k值不宜取得过大。事实上，不同的k-mer在描述DNA序列方面有不同的影响，正如文献 [15, 16]中所说的，在2到7的范围里,k=4是最适合表示DNA序列的，因此我们使用k=4 来表示元基因序列。具体的说，我们滑动一个长为4 的窗口来统计一个序列中k-mer的频率，他的互补序列也同样被考虑，因此一个序列的维度是256。

2.2 基于主题模型的特征向量空间变换

元基因组中，序列被当成文档，k-mer被当成关键词，来自同一物种的序列应该比不同物种的序列有更多相似的主题信息，因此，对于目前考虑的归类问题，主题信息在描述元基因组序列方面可能比k-mer更有效。我们采用隐含狄利克雷分布(LDA)–一个机器学习领域非常流行的主题来处理这个问题。在上一章中，我们也对主题模型进行了介绍。用LDA对序列建模之后，我们可以得到每个序列的主题分布。图2.2表示LDA在元基因组序列上的运用，左层图表示DNA序列，中间层表示主题，右层图表示k-mer，我们用每个序列的主题分布来表示序列。由于主题的数量通常小于k-mer的数量，这个过程等价于降维。此处，主题个数是一个可调节参数。在我们实验研究中，我们对于模拟数据和真实数据分别设置主题数为20和100。

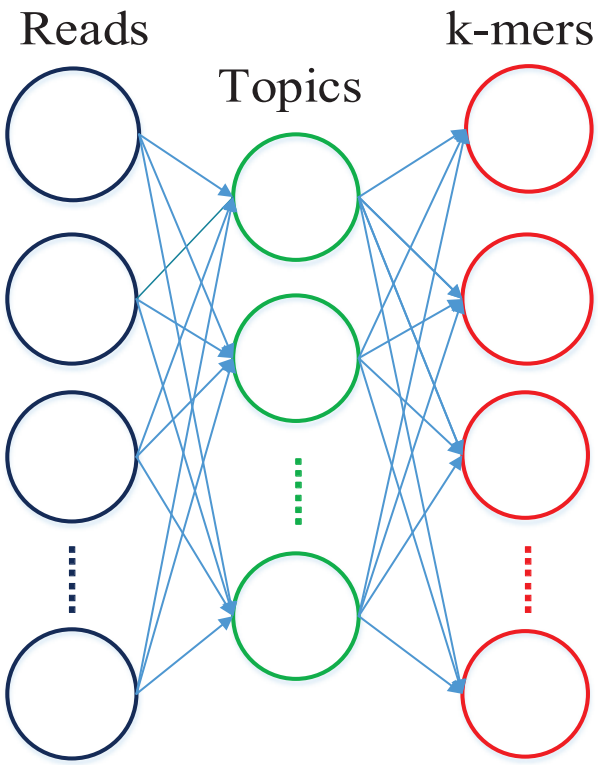

图 2.2: 主题模型在元基因组序列上的应用

2.3 基于SKWIC算法的元基因组序列聚类

和MCluster [20]一样, 我们采用SKWIC算法对向量化的元基因序列进行聚类。SKWIC是在经典的k-means算法基础上增加了自动加权机制的聚类算法

[21]。k-means或者k-median方法是现有的非监督学习方法，在这两个方法中，特征是等同对待的。然而，在聚类的过程中，它并没有挖掘类和特征集的关系。为了解决这个问题，Frigui等人提出了SKWIC算法，它是k-means算法的变形。粗略的说，它通过挖掘了k-mer偏好性，来提高聚类的性能。而且在miRNA序列聚类[32]以及元基因组归类问题[20]中，k-mer偏好性能够提高生物序列的聚类性能的说法, 也同样得到验证。

SKWIC算法试图最小化下面的目标函数:

$$J(K,V;\chi)=\sum_{i=1}^{K}\sum_{x_j\in\chi_i}\sum_{k=1}^{n}v_{ik}D^k_{wc_{ij}}+\sum_{i=1}^{K}\delta_i\sum_{k=1}^{n}v_{ik}^2 \tag{2.1}$$

subject to

$$v_{ik}\in[0,1]\quad\forall i,k\quad and\quad\sum_{k=1}^{n}v_{ik}=1,\quad\forall i \tag{2.2}$$

K 是类的个数, n 是特征的个数, 这里是主题的个数。X_i 是第i个类的序列个数, v_{ik} 是第i个类在第k 个特征上的权值, $D^k_{wc_{ij}}$ 是第j条序列和第i个类中心在第k维特征上的距离。在这个目标函数中，我们应该选择一种距离度量方式。根据[20]中的结论，和欧式距离以及余弦距离相比，曼哈顿距离在对生物序列进行聚类时效果最佳，因此此处我们也采用曼哈顿距离进行距离度量。

与传统的K-means 算法不同，目标函数(2.1)额外考虑了每一维特征对于每一个类的权重v_{ik}。δ_i 来权衡v_{ik} 的相对重要性。为了解决这个优化问题，我们采用拉格朗日乘子法，

$$v_{ik}=\frac{1}{n}+\frac{1}{2\delta_i}\sum_{x_j\in\chi_i}[\frac{\sum_{l=1}^{n}D^l_{wc_{ij}}}{n}-D^k_{wc_{ij}}] \tag{2.3}$$

上述等式中的参数δ_i是非常重要的，因为它被用来权衡等式2.1。如果δ_i太小, 等式2.1第二部分的贡献将会被忽视，并且类的某一个维度会有相对较大的权重，其他维度可能会有非常小甚至接近0的权重。另一方面，如果δ_i如果太大，每个cluster 的所有维度都会接近$\frac{1}{n}$

δ_i 计算公式如下：

$$\delta_i^{(t)}=K_\delta\frac{\sum_{x_j\in\chi_i^{(t-1)}}\sum_{k=1}^{n}v_{ik}^{(t-1)}(D^k_{wc_{ij}})^{(t-1)}}{\sum_{k=1}^{n}(v_{ik}^{(t-1)})^2}. \tag{2.4}$$

SKWIC的聚类过程就是重复下面的步骤直到类中心不变或者变化范围足够小：

1. 设定聚类个数为K (序列所属的物种数);

2. 随机选择K 个类的类中心, 并将权值矩阵设定为v_{ik}=$\frac{1}{n}$;

3. 利用公式 2.3更新权值矩阵v_{ik};

4. 将序列归到距离它最近的类;

5. 更新每个类的类中心;

6. 利用公式 2.4更新δ_i

2.4 数据集

2.4.1 模拟数据

数据由MetaSim[33] (一个基因组和元基因组序列模拟软件)生成。我们从物种丰度各异的物种中采样出元基因组序列数据。具体的，我们从NCBI下载了多种微生物的数据库，然后选择了其中的十种物种的基因组采样，生成模拟数据，十个物种名如下：*Pseudomonas_aeruginosa_PAO1, Marinobacter_sp._BSs20148, Chromohalobacter_salexigens_DSM_3043, Legionella_pneumophila_str, Nitrosococcus_oceani_ATCC_19707, Cycloclasticus_sp._P1, Salmonella_typhimurium_LT2, Xanthomonas_oryzae_pv._oryzae_KACC10331,*
Aeromonas_salmonicida_subsp._salmonicida_A449, Vibrio_cholerae_O395

由于MetaCluster 3.0 在长序列上表现良好，我们模拟了不同物种丰度和物种个数(2,3,4,5,10)的长序列数据,序列平均长度设定为1000个碱基，序列个数为5k，16个数据集表示为D1 到D16，由于AbundanceBin是专门处理高丰度的序列，我们同样生成了相对高丰度的序列(50k和500k序列)，序列平均长度1000bp，物种名分别为2,3,5,10种。10 个数据集分别称为S1 到S10。这26个数据集的详细说明在表2.1 和表2.2 中。

在真实数据集中，包含数以百万的短序列是非常常见的，因此我们也模拟了两个数据集，分别有一百万条平均长度为75bp的序列，分别包含20和50个物种，称为数据集A 和B。数据集A中包含20个物种中，相对测序深度为1、3、5和10的物种数各为5。数据集B中包含了50个物种，有6个物种相对测序深度为6, 5个物种相对测序深度为8,5个物种相对测序深度为10，剩下的物种相对测序深度为1，数据集A和B的详细说明在表2.3中。

由于MetaCluster 5.0只能处理极高丰度的短序列，我们也生成了5个数据集，分别有3000k个长为128bp的短序列，并且将我们的方法与MetaCluster 5.0 进行比较。这些数据集分别称为C、D、E、F、G，数据集的详细说明在2.4中。

表 2.1: 低丰度的模拟数据(序列平均长度1000bp)。

数据集	序列数	物种数	丰度比
D1	5k	2	1:1
D2	5k	2	1:2
D3	5k	2	1:4
D4	5k	2	1:6
D5	5k	2	1:8
D6	5k	2	1:10
D7	5k	2	1:12
D8	5k	3	1:1:1
D9	5k	3	1:3:9
D10	5k	4	1:3:3:9
D11	5k	5	1:1:1:1:1
D12	5k	5	1:1:3:3:9
D13	5k	10	1:1:1:1:1:1:1:1:1:1
D14	50k	3	1:3:9
D15	50k	4	1:3:3:9
D16	50k	5	1:1:3:3:9

表 2.2: 相对高丰度的模拟数据(序列平均长度1000bp)。

数据集	序列数	物种数	丰度比
S1	50k	2	1:1
S2	50k	3	1:1:1
S3	50k	3	1:3:9
S4	50k	5	1:1:3:3:9
S5	50k	10	1:1:1:1:1:1:1:1:1:1
S6	500k	2	1:1
S7	500k	3	1:1:1
S8	500k	3	1:3:9
S9	500k	5	1:1:3:3:9
S10	500k	10	1:1:1:1:1:1:1:1:1:1

表 2.3: 极高丰度模拟数据集(平均序列长度是75bp)。

数据集	序列数	物种数	丰度比
A	1million	20	1 X 5:3 X 5:5 X 5:10 X 5
B	1million	50	1 X 34:6 X 6:8 X 5:10 X 5

表 2.4: 极高丰度模拟数据 (序列平均长度是128bp)。

数据集	序列数	物种数	丰度比
C	3000k	2	1:1
D	3000k	3	1:1:1
E	3000k	3	1:3:9
F	3000k	5	1:1:3:3:9
G	3000k	10	1:1:1:1:1:1:1:1:1:1

2.4.2 真实数据集

鉴于研究人员已经对NCBI的Acid Mine Drainage元基因组数据[34]进行了大量研究，我们也采用该数据集作为真实数据集来评价我们的方法。这个真实数据集包含2534 个重叠群(contig), 序列长度为5000bp，这些重叠群是由103462个高质量的修整过的短序列组装而成的。数据集包括5个已知的物种：*Leptospirillum sp.Group II, Leptospirillum sp.Group III, Ferroplasma acid armanus Type I, Ferroplasma sp.Type II and Thermoplasmatales archaeon Gpl* 以及一些来自未知物种的序列。五个物种分别属于两个超界和三个属，分类图如图 2.3所示。序列有2534 个contig。对于包含了未知物种信息序列的数据集，对聚类算法的结果进行评估比较困难，我们删除其中没有物种注释的序列，并得到2424 个contigs，表示为数据集$R1$。

2.4.3 评价标准

为了评价聚类结果，我们考虑三种度量方法，Precision(Pr)，Sensitivity(Se)以及F1-measure(F1)。假定有一个元基因组数据集包含了N个物种，并最终归到M个类中，R_{ij}表示第i个Cluster中包含第j个物种的序列的数量。

Precision和Sensitivity的定义如下所示

$$Pr = \frac{\sum_{i=1}^{M} \max_j(R_{ij})}{\sum_{i=1}^{M}\sum_{j=1}^{N} R_{ij}}, \tag{2.5}$$

$$Se = \frac{\sum_{j=1}^{N} \max_i(R_{ij})}{\sum_{i=1}^{M}\sum_{j=1}^{N} R_{ij} + \text{number of unclassified reads}} \tag{2.6}$$

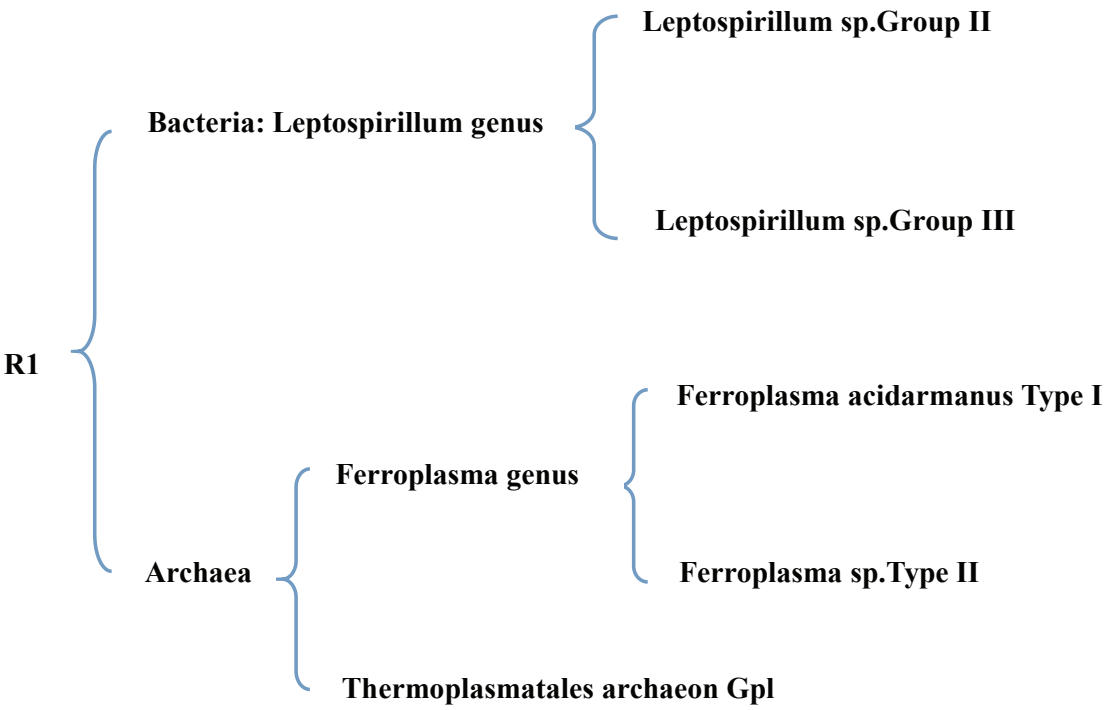

图 2.3: 数据集R1的物种分类图。

上述"unclassified reads" 表示聚类算法未进行归类的序列。F1-measaure定义如下所示：

$$F1 = \frac{2 * Pr * Se}{Pr + Se} \tag{2.7}$$

2.5 实验结果

2.5.1 主题个数的影响

概率主题模型是一种非监督技术。模型中的主题信息是隐藏的，所以我们需要为每个数据集设定主题个数。此处，我们检测LDA模型中主题个数如何影响TM-MCluster 的聚类性能。我们采用D12数据集，序列来自5个物种，并且修改主题的个数，使之从2到逐渐变化到100，然后观测我们提出算法的性能。图2.4 的结果显示，当主题个数是20 的时候，我们的方法可以取得较好的聚类效果。当主题个数为2时，聚类结果差强人意。显然，太少的主题个数可能会导致信息的丢失，太多的主题个数也许会引入噪音，也会影响聚类效果。

2.5.2 模拟数据集实验结果

首先，将我们的方法与MetaCluster3.0 和MCluster在4个均匀分布的数

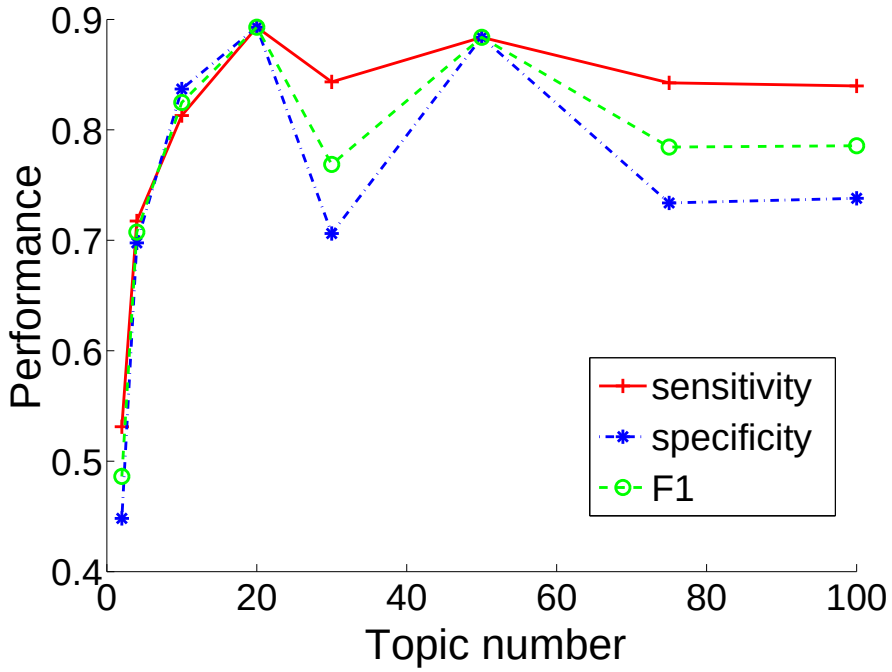

图 2.4: 主题个数对TM-MCluster聚类效果的影响。

据集上的结果进行比较，数据集D1、D8、D11以及D13分别包含2、3、5以及10个物种。结果显示在表2.5 中。 从表2.5中可以看出，我们的方法在D1、

表 2.5: 在相同丰度比的模拟数据集(D1、D8、D11和D12)上的聚类结果。加粗数值表示某个数据集上的最好结果。

Dataset	MetaCluster 3.0			MCluster			TM-MCluster		
	Pr	Se	F1	Pr	Se	F1	Pr	Se	F1
D1	**.9989**	.9628	.9805	.9877	.9877	.9877	.9882	**.9882**	**.9882**
D8	.7432	.9218	.8229	.9158	.9158	.9158	**.9586**	**.9586**	**.9586**
D11	.8215	.8766	0.8481	**.9002**	**.9002**	**.9002**	.8394	.8394	.8394
D13	.4335	**.8732**	.5794	.706	.6894	.6976	**.7574**	.7732	**.7652**

D8和D11这三个数据集上取得最高的F1值，在D8和D13这两个数据集上取得最高的Precision，在D1和D8这两个数据集上取得最高的Sensitivity。结果表明我们的方法在丰度比均匀的数据集上聚类性能优异。

我们同样分析TM-MCluster在12个非均匀分布的数据集上的性能，结果显示在表2.6。在12个测试数据集中，我们的方法分别在10，6和5个数据集上取得了最高的F1、Precision以及Sensitivity。MCluster在三个数据集上取得最好的F1值和sensitivity, 而MetaCluster3.0分别在7个和5个数据集上分别取得最高

的Precision 和Sensitivity，但是F1值却差强人意。值得注意的是，我们的方法似乎在分布不均匀的数据集上表现更加优异。

表 2.6: 12个非均匀丰度比数据集上的聚类结果。加粗数值表示某个数据集上的最好结果。

Dataset	MetaCluster 3.0			MCluster			TM-MCluster		
	Pr	Se	F1	Pr	Se	F1	Pr	Se	F1
D2	**.9997**	.9648	.9820	.9888	**.9888**	**.9888**	.9860	.9860	.9860
D3	**.9998**	.9596	.9793	.9950	**.9950**	**.9950**	.9948	.9948	.9948
D4	**1.0000**	.9612	.9802	.9942	.9942	.9942	.9946	**.9946**	**.9946**
D5	**1.0000**	.9608	.9800	.9950	.9950	.9950	.9954	**.9954**	**.9954**
D6	**1.0000**	.9610	.9801	.9966	**.9966**	**.9966**	.9966	**.9966**	**.9966**
D7	**1.0000**	.9618	.9805	.9980	.9980	.9980	.9988	**.9988**	**.9988**
D9	.7277	**.9628**	.8289	.8974	.8974	.8974	**.9320**	.9320	**.9320**
D10	.7345	.9096	.8127	.8852	.8852	.8852	**.9156**	**.9156**	**.9156**
D12	.7489	**.9066**	.8202	.8524	.8524	.8524	**.8930**	.8930	**.8930**
D14	.7275	**.9539**	.8255	.8863	.8860	.8863	**.9420**	.9420	**.9420**
D15	.7472	**.9202**	.8247	.8764	.8764	.8765	**.9070**	.9070	**.9070**
D16	.6792	**.9106**	.778	.8546	.8546	.8546	**.8875**	.8875	**.8875**

真实数据集中包含了大量高丰度序列，因此在高丰度数据集上的表现更能体现算法的实际价值。我们将TM-MCluster与AbundanceBin 和MCluster 进行比较，结果如表2.7所示。在10个测试数据集上，我们的方法分别在9、6和8个数据集上取得最高的F1值，Sensitivity 以及Precision，表现最稳定。MCluster 只在数据集S6取得最高的F1值，在数据集S6和S10上取得最高的Precision，AbundanceBin 在4个数据集上取得最高的Sensitivity。综合来看，在较高丰度数据集上，我们方法性能优于其他方法。

现实中，越来越多的元基因组数据都是短序列(几百bp)，因此我们通过处理短序列数据来评价我们的方法。由于MetaCluster 3.0 处理短序列能力有限，我们只列出AbundanceBin、MCluster以及我们的方法在数据集A和B(短序列)的性能。结果如表2.8所示和AbundanceBin以及MCluster相比，TM-MCluster 取得了最高的F1 值以及Precision，这和我们方法在长序列上的结果是不谋而合的。

由于对于大规模的元基因组数据进行聚类耗时费内存，时间和空间效率也是一项重要的评价指标。我们列出AbundanceBin MCluster以及我们的方法在数据集A 和B 上的时间和空间消耗，结果显示在表3.1中。我们可以看出AbundanceBin花费最少的内存，而MCluster运行最快。由于训练LDA是耗时的，TM-MCluster 花费最多的时间，此外空间消耗也较大。

表 2.7: 高丰度数据集上的聚类结果。加粗数值表示某个数据集上的最好结果。

Dataset	**AbundanceBin**			**MCluster**			**TM-MCluster**		
	Pr	Se	F1	Pr	Se	F1	Pr	Se	F1
S1	.7258	.9740	.8317	.9875	.9875	.9875	**.9882**	**.9882**	**.9882**
S2	.4047	.9405	.5600	.9154	.9154	.9154	**.9519**	**.9519**	**.9519**
S3	.5866	.7528	.6594	.8873	.8873	.8873	**.9361**	**.9361**	**.9361**
S4	.4106	**.9441**	.5723	.8554	.8554	.8554	**.8921**	.8921	**.8921**
S5	.1748	**.9871**	.2970	.7361	.7241	.7301	**.7578**	.7546	**.7562**
S6	.7266	**.9999**	.8416	**.9873**	.9873	**.9873**	.9869	.9869	.9869
S7	.3991	**.9999**	.5705	.9173	.9173	.9173	**.9545**	.9545	**.9545**
S8	.8591	.8591	.8591	.8868	.8868	.8868	**.9393**	**.9393**	**.9393**
S9	.6457	.6476	.6466	.8581	.8581	.8581	**.8880**	**.8880**	**.8880**
S10	.1888	.7223	.2993	**.7253**	.7161	.7207	.7196	**.7317**	**.7256**

表 2.8: AbundanceBin，MCluster以及TM-MCluster在短序列数据集上(数据集A，B)的聚类性能。加粗数值表示最好的Precision，Sensitivity和F1 值。

Dataset	**AbundanceBin**			**MCluster**			**TM-MCluster**		
	Pr	Se	F1	Pr	Se	F1	Pr	Se	F1
A	.2270	.9878	.3692	.2250	**1.0000**	.3674	**.3165**	.6471	**.4251**
B	.0757	.9878	.1407	.0744	**1.0000**	.1384	**.1338**	.5836	**.2177**

表 2.9: AbundanceBin、MCluster以及TM-MCluster在短序列数据集(数据集A和B)上的内存和时间消耗。

Dataset	**AbundanceBin**		**MCluster**		**TM-MCluster**	
	Memory	Time	Memory	Time	Memory	Time
A	3.07GB	2.15h	3.20GB	1.36h	4.12GB	3.11h
B	3.20GB	3.20h	3.46GB	2.38h	4.10GB	3.31h

最后，我们还比较了TM-MCluster以及MetaCluster 5.0在数据集C、D、E、F和G上的性能，结果列在表2.10上。结果表明，TM-MCluster在4个数据集上取得明显高于MetaCluster 5.0的Sensitivity，这主要是由于MetaCluster 5.0在聚类时，将低丰度物种的序列都归到小的类里，最后再丢弃。不过，MetaCluster5.0在五个数据集上都取得较高的Precision。由于F-measure是Sensitivity和precision上的权衡，我们的方法依然在4个数据集上取得了较高的F-measure。此外，MetaCluster 5.0在数据集D差强人意，而这个数据碰巧有最大数量的物种数以及多种多样的丰度比。总而言之，在短序列的高丰度数据集上，我们的方法取得优于MetaCluster 5.0的性能。

表 2.10: TM-MCluster和MetaCluster 5.0的性能比较。

Dataset	**MetaCluster 5.0**			**TM-MCluster**		
	Pr	Se	F1	Pr	Se	F1
C	**.9944**	.3862	.5563	.9793	**.9793**	**.9793**
D	**.9904**	.4290	.5986	.7198	**.7198**	**.7198**
E	**.9770**	**.4806**	**.6437**	.6923	.4645	.5574
F	**.9770**	.3178	.4796	.5801	**.4645**	**.5159**
D	**.8662**	.0066	.0131	.2141	**.7988**	**.3377**

2.5.3 真实数据集实验结果

此外，我们还在真实数据集上测试我们方法的性能。从图2.11上，我们知道R1上的序列属于两个超界，三个属以及五个物种，我们考虑按照物种分类的不同层次来进行聚类。因此我们预先设定AbundanceBin MCluster以及我们的方法的聚类个数分别为2，3和5。由于MetaCluster 3.0可以自动决定最终聚类的个数，我们不用为它设定类的个数。对于我们的方法，主题个数设定为100，最终MetaCluster 3.0 输出了2个类。上述所有结果都显示在表2.11中。尽管MetaCluster 3.0可以自动决定类的个数，他的结果是不准确的，因为R1数据集中有5个物种。对于其他三个方法，AbundanceBin 取得最高的Sensitivity，但是Precision是最低的。对于每个预先设定的聚类个数，我们的方法取得最高的F1值。值得一提的是，当预设类个数为数据集实际物种个数5 时，我们的方法取得最高的Precision以及F1值，以及仅次于AbundanceBin的Sensitivity。

对于AbundanceBin，MCluster以及我们的方法，当预设的聚类个数从2上升到5时，聚类性能出现下降趋势。这是因为，设定聚类个数分别为2、3和5时，会将R1 数据集分别往界、属和类的层次聚类。在一个更高层面的，两个类中心的距离一般来说是大于低层次的类中心，因此再较高层次上更容易聚类。

表 2.11: 真实数据集R1上的聚类结果。

Methods	# **Cluster**	**Pr**	**Se**	**F1**
MetaCluster 3.0	2	**.7328**	.8441	.7845
AbundanceBin	2	.3952	**.9934**	.5655
	3	.3952	**.9934**	.5655
	5	.3952	**.9893**	.5648
MCluster	2	.7050	.9422	.8066
	3	.7054	.9179	.7978
	5	.6972	.6444	.6698
TM-MCluster	2	.7186	.9682	**.8250**
	3	**.7211**	.9645	**.8252**
	5	**.7182**	.9130	**.8040**

2.6 小结

本章中，我们详细的描述我们的算法，首先用n-gram模型来进行特征提取，然后运用LDA模型将序列从k-mer空间映射到主题空间，最后和MCluster类似，运用SKWIC算法对主题向量表示的序列进行聚类。我们在模拟数据和真实数据集上，对我们提出的方法进行了全方面的评价，并且与已有的MetaCluster 3.0 /5.0、AbundanceBin 和MCluster 进行了比较分析，结果显示我们的方法在多数数据集上聚类性能都优于上述方法。

第三章 基于深度学习的元基因组聚类算法

3.1 深度学习概述

深度学习是机器学习的一块新的研究领域，它让机器学习更靠近人工智能的目标。与传统机器学习算法不同，深度学习作为表征学习的一种，在学习的过程中，不需要手工设计数据的特征，而是利用多层神经网络的深层结构，由浅层到深层自动学习对目标有利的数据特征表示，另外，深度学习算法学习到的数据特征表示能够获得数据的内部结构，对于不同的学习任务(例如：分类、回归等)，这种表示都可以使用。而机器学习的特征设计大都是基于特定任务的，不同的学习任务，设计出来的特征不同。因此深度学习是一种新型的基于多层神经网络结构的学习算法，它与传统人工神经网络的训练不同，BP算法作为传统多层神经网络的经典训练算法，在多层网络的训练过程中结果很不理想[35]。Bengio 等人[36, 37]基于深度信念网（DBN）提出无监督逐层训练算法，为解决深层结构相关优化难的问题带来希望。LeCun 等人[38]提出的卷积神经网络（CNNs）是第一个真正多层结构学习算法，它利用空间相对关系减少参数数目以提高BP训练性能。

数据从输入到输出可以用一个流向图(Flow Graph)来表示[35]，流向图的每个节点表示计算的一步和该步计算得到的值。考虑一个计算集允许每个节点和可能的图结构，并且定义了一个函数族。输入节点没有子节点，输出节点没有父节点。

这种流向图的一个特别属性是深度(depth): 从输入到输出的最长路径的长度。

传统的前馈神经网络能够被看作拥有等于层数的深度，SVM的深度为2（一个对应于核输出或者特征空间，另外一个对应着输入的线性组合）。借助深度学习的算法，人类对“抽象概念”的处理有了解决方法。在技术手段方面，有云计算对大数据的并行处理能力。

深度学习的动机在于建立、模拟人脑进行分析学习的神经网络，它模仿人脑的机制来解释数据[35]，例如图像，声音和文本。传统神经网络的训练算法的缺点[39]：

1. 容易出现过拟合，参数很难调节。
2. 训练速度较慢，当神经网络的层数太多时（大于7层），残差传播到最前面的网络层会变得很小，出现梯度扩散的现象。
3. 收敛到局部最小值
4. 只能使用有标签数据来训练，但是实际应用中数据大部分是无标签的。

卷积神经网络(CNN)作为第一个真正成功训练多层网络结构的学习算法，与DBN不同，它属于判别型模型，卷积结构的使用也是深度学习在语音、图像和自然语言处理中取得成功的关键因素。卷积神经网络由卷积层和次抽样层组成，其隐藏层的单元有一个时间或者空间位置且只与特定窗口的原始输出的值有关系。CNN作为深度学习框架是基于最小化预处理数据要求而产生的。受早期的时延神经网络影响，CNN靠共享时域权值降低复杂度。CNN是利用空间关系减少参数数目以提高一般前向BP训练的一种拓扑结构，在CNN中被称做局部感受区域的图像的一小部分作为分层结构的最底层输入。信息通过不同的网络层次进行传递，因此在每一层能够获取对平移、缩放和旋转不变的观测数据的显著特征。

之后，不断有新的深层网络结构的提出，和新的训练技巧的提出，如栈式自动编码机，层叠受限玻尔兹曼机，深度玻尔兹曼机，卷积深度信念网等新型网络结构，以及Dropout，Maxout，稀疏性(Sparsity)，权值衰减(Weight-decaying)，去噪(denoising)等技巧的提出，极大地丰富了深度学习算法及其学习框架，使得深度学习的应用也不再局限于语音和图像方面，在自然语言处理[40]，时间序列数据建模[41] 方面都取得了很好的成果。

深度学习算法分为有监督学习和无监督学习两种，卷积神经网络属于有监督学习，而深度信念网和自动编码机则属于无监督学习。更严格的讲，对于要解决的实际问题，使用无监督学习算法进行预训练神经网络的权值，然后用来初始化深层神经网络这样网络的性能就会极大提升[42]。对于无监督预训练算法，自动编码机在概念上比较简单，但是受限玻尔兹曼机模型对于不同的参数或者不同类型的可视单元和隐藏单元有不同的模型。因此近年来，对于自动编码机的研究偏重于正则化（稀疏、去噪等），而受限玻尔兹曼机不同，它是一种生成模型(Generative Model)，可以抽取样本。

3.2 自动编码机

自动编码机属于无监督学习的一种，因为在计算机视觉，语音处理和自然语言处理领域，传统的有监督学习需要人工设计特征，如果设计的特征比较好，

那么有监督学习的算法就很有效，但是特征设计是一件费时费力的工作，而且这种特征对于特定问题可以很好地解决，对于其他问题就需要重新设计特征。

自动编码机学习的目标值等于输入值，它学习到的函数是一个恒等函数[43]，它的意义在于学习输入数据的一种表示，这种表示可以重构输入数据，这种表示就是特征。它的结构示意图见图3.1:

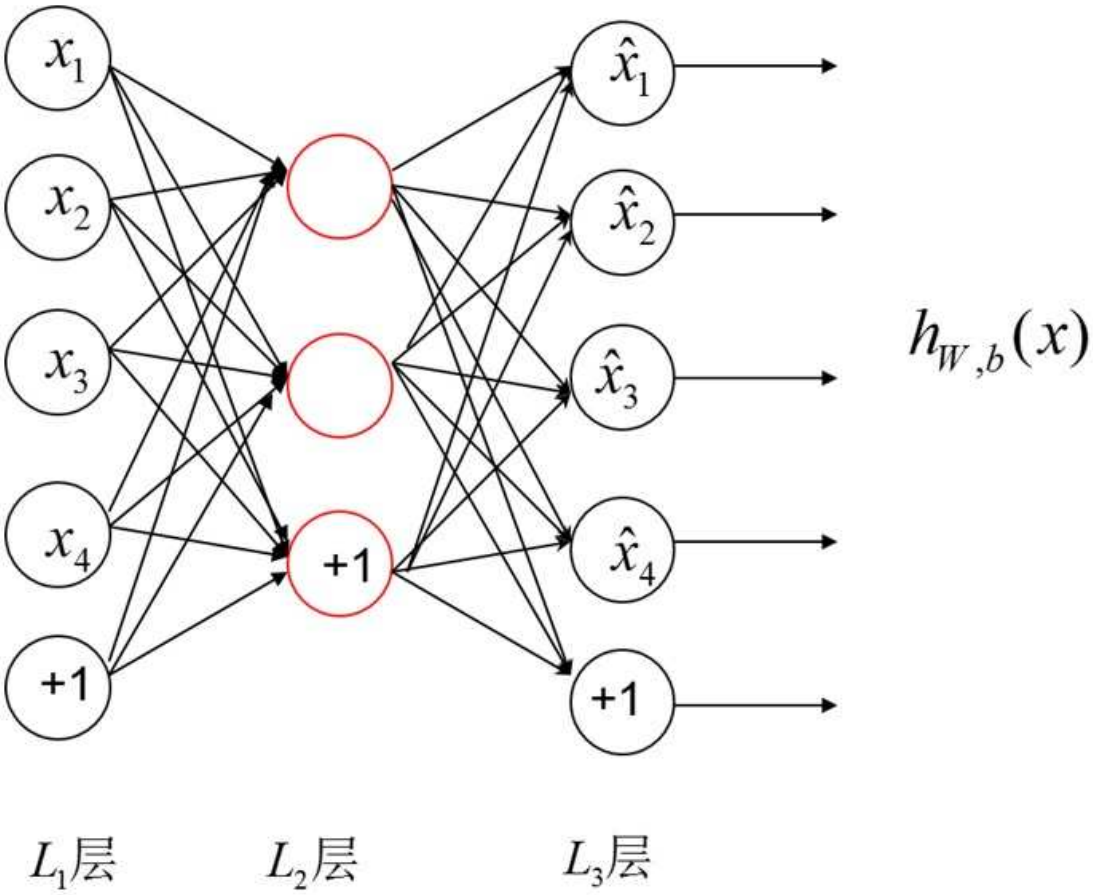

图 3.1: 自动编码机的结构

设自动编码机的输入为$x\epsilon[0,1]^d$，到隐藏层的第一层映射（也称为编码机）得到的隐含表示$y\epsilon[0,1]^d$

$$y = s(Wx + b) \tag{3.1}$$

从隐含表示y映射到输出z来重构输入（也称解码机），z和x的尺寸大小相同。

$$z = s(W'y + b') \tag{3.2}$$

这里并不表示转置，z可以看作为x的预测。W'一般约束为W的转置，这样可以减少训练的参数，但是不加这个约束也可以训练自动编码机。

该模型的参数有W, b, b'(如果不加约束项，还包括W'),优化目标是平均重构误差最小。重构误差可以用很多评定方法，具体选择与输入数据的分布假设有

关[44]。一般使用传统的方差测评$L(x,z)=\|x-z\|^2$，或者如果输入数据是二值向量，可以使用交叉熵来测评：

$$L_H(x,z)=-\sum_{k=1}^{d}[x_k log z_k+(1-x_k)log(1-z_k)] \tag{3.3}$$

我们希望得到的中间表示y是输入x的分布式表示，当隐含单元数目较小时，自动编码机被迫去学习输入数据的压缩表示，y一般是x的有损压缩，但是如果输入是完全随机的，这个压缩表示将很难学习到，如果输入数据本身具有某种隐藏的特定结构，那么自动编码机就会发现输入数据中的某些相关性，从而将数据输入用低维表示。当隐含单元数目较大时，需要给自动编码机增加一些限制条件来发现输入数据的结构，否则学习到的将是无用的信息，具体地，如果给隐藏神经元加入稀疏性限制，那么自动编码机将会学习到输入数据中的一些有趣的结构。

令人惊讶的是，当隐含单元个数大于输入数据单元时，非线性自动编码机可以学习到更加有用的表示[45]。较为合理的解释是，使用随机梯度下降法训练自动编码机，提前终止迭代与对参数加$L2$规范项类似。

一个标准的自动编码机的学习算法为算法1(一个隐藏层的自动编码机)：

Algorithm 1 自动编码机学习算法

Input:

k-mer特征表示的序列

Output:

自动编码机学习到的特征表示的序列

1: 采用前向传播, 计算L_2, L_3层的激活值;

2: 计算输出层L_3的每个单元i的误差值:

$$\delta^{(3)}=-(y-z)\bullet f^{'}(s^{(3)}) \tag{3.4}$$

3: 对于l=2, 1 设

$$\delta^{(l)}=((W^{(l)})^T\delta^{(l+1)})\bullet(f'(s^{(l)})) \tag{3.5}$$

4: 计算损失函数关于参数的偏导数

$$\frac{\partial J(W,b;x,y)}{\partial W^{(l)}}=\delta^{(l+1)}(a^{(l)})^T \tag{3.6}$$

$$\frac{\partial J(W,b;x,y)}{\partial b^{(l)}}=\delta^{(l+1)} \tag{3.7}$$

5: 使用L-BFGS算法求解神经网络的参数

6: 更新参数

基于深度学习的聚类算法也同样包含三步：1) 用k-mer频率向量来表示序列 2) 用自动编码机编码特征 3) 采用SKWIC 算法对向量化的序列进行聚类。图3.2中显示了工作流程。

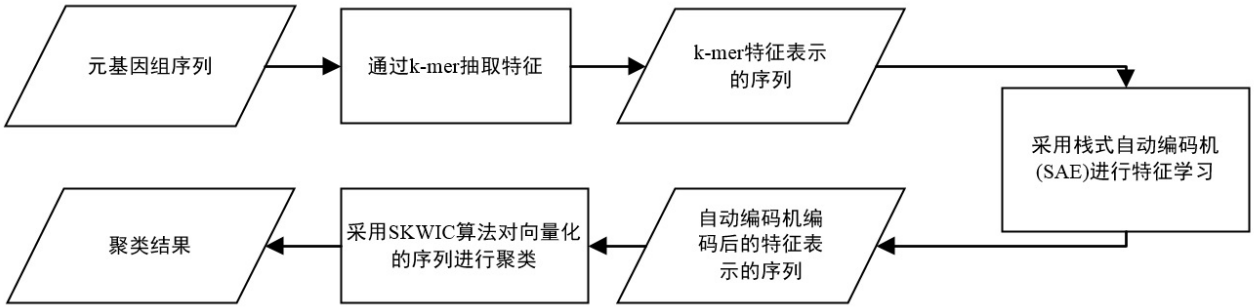

图 3.2: 基于深度学习的元基因组聚类算法的工作流程。

3.3 实验结果

我们采用上一章实验的数据集。由于深度学习对于较大的数据集效果较好，因此我们采用500k, 1000k和3000k的序列进行特征学习。具体的，500k的数据集S5、S6、S7、S8、S9 和S10，1000k的数据集A、B，3000k的数据集为C、D、E、F和G。多层自动编码机可以学习到更好的特征，经过参数调节，我们选择7层的自动编码机进行特征学习，采用4-mer, 输入特征为256，输出的特征数为512，中间5层神经网络的特征个数依次为1024、1024、512、1024和1024，也即采用了3个自动编码机进行特征学习，然后同样采用SKWIC算法对学习到的特征进行聚类，实验结果如表3.1：

表 3.1: 采用自动编码机和SKWIC算法在12个数据集上的聚类效果。

Dataset	**AutoEncoder**		
	Pr	Se	F1
S6	.9889	.9889	.9889
S7	.9022	.9022	.9022
S8	.8450	.8450	.8450
S9	.8292	.8292	.8292
S10	.7216	.7150	.7183
A	.4041	.2337	.2961
B	.2123	.3129	.2530
C	.8192	.8192	.8192
D	.6192	.6192	.6192
E	.9805	.9805	.9805
F	.7784	.9772	.8665
G	.3028	.7932	.4383

3.4 小结

本章中，我们介绍了目前非常流行的深度学习，对元基因组数据进行归类是非监督学习任务，采用自动编码机进行特征学习，然后采用SKWIC算法进行聚类。实验结果也表明，基于深度学习的方法也取得了不错的聚类效果。

第四章 总结与展望

在这篇文章里，我们提出了一个新的方法TM-MCluter对元基因组序列进行聚类。新方法结合了k-mer，主题模型以及自动加权的聚类方法来提高元基因数据的聚类性能。我们用大量的模拟和真实数据来评价我们的方法，并且我们的猜想得到验证，TM-MCluster优于现有的AbundanceBin、MetaCluster 3.0/5.0以及最近的MCluster 方法。实验结果表明，采用主题模型可以有效提高元基因组序列的聚类性能。

此外，我们还采用了当前最热的深度学习进行了探索，用自动编码机进行特征学习，然后采用SKWIC算法进行聚类，也取得了令人满意的效果。

元基因组序列分析一直是研究热点。如何提高聚类的效果，聚类问题相对于分类来说更加困难，此外对于大规模数据的处理也是一个难点。

通过文献阅读，我们了解到有研究人员曾经采用Ramanujan Fourier变换(后面简称为RFT)对DNA序列进行特征提取[46] 虽然对病毒的DNA序列进行层次聚类，因此也考虑用RFT对序列进行特征表示，但是实际的聚类性能差强人意，无论用k-means，SKWIC等聚类算法都效果不好。

参考文献

[1] 刘莉扬, 崔鸿飞, 田埂: 高通量测序技术在宏基因组学中的应用. 中国医药生物技术**8**(3) (2013)

[2] Qin, J., Li, R., Raes, J., Arumugam, M., Burgdorf, K.S., Manichanh, C., Nielsen, T., Pons, N., Levenez, F., Yamada, T., *et al.*: A human gut microbial gene catalogue established by metagenomic sequencing. Nature **464**(7285), 59–65 (2010)

[3] Khachatryan, Z.A., Ktsoyan, Z.A., Manukyan, G.P., Kelly, D., Ghazaryan, K.A., Aminov, R.I.: Predominant role of host genetics in controlling the composition of gut microbiota. PloS One **3**(8), 3064 (2008)

[4] Metagenomics. `http://www.decodegenomics.com/product-service/product-service131.html`

[5] Mavromatis, K., Ivanova, N., Barry, K., Shapiro, H., Goltsman, E., McHardy, A.C., Rigoutsos, I., Salamov, A., Korzeniewski, F., Land, M., *et al.*: Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. Nature Methods **4**(6), 495–500 (2007)

[6] 聂鹏宇, 潘玮华, 徐云: 基于仿射聚类的宏基因组序列物种聚类. 计算机系统应用**11** (2013)

[7] Huson, D.H., Richter, D.C., Mitra, S., Auch, A.F., Schuster, S.C.: Methods for comparative metagenomics. BMC Bioinformatics **10**(Suppl 1), 12 (2009)

[8] McHardy, A.C., Martin, H.G., Tsirigos, A., Hugenholtz, P., Rigoutsos, I.: Accurate phylogenetic classification of variable-length dna fragments. Nature Methods **4**(1), 63–72 (2006)

[9] Stark, M., Berger, S., Stamatakis, A., von Mering, C.: Mltreemap-accurate maximum likelihood placement of environmental dna sequences into taxonomic and functional reference phylogenies. BMC Genomics **11**(1), 461 (2010)

[10] Diaz, N.N., Krause, L., Goesmann, A., Niehaus, K., Nattkemper, T.W.: Tacoa–taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. BMC Bioinformatics **10**, 56 (2009)

[11] Brady, A., Salzberg, S.L.: Phymm and phymmbl: metagenomic phylogenetic classification with interpolated markov models. Nature Methods **6**(9), 673–676 (2009)

[12] Wu, Y.-W., Ye, Y.: A novel abundance-based algorithm for binning metagenomic sequences using l-tuples. Journal of Computational Biology **18**(3), 523–534 (2011)

[13] Yang, B., Peng, Y., Leung, H., Yiu, S.-M., Qin, J., Li, R., Chin, F.Y.: Metacluster: unsupervised binning of environmental genomic fragments and taxonomic annotation. In: Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology, pp. 170–179 (2010). ACM

[14] Leung, H.C., Yiu, S.-M., Yang, B., Peng, Y., Wang, Y., Liu, Z., Chen, J., Qin, J., Li, R., Chin, F.Y.: A robust and accurate binning algorithm for metagenomic sequences with arbitrary species abundance ratio. Bioinformatics **27**(11), 1489–1495 (2011)

[15] Chor, B., Horn, D., Goldman, N., Levy, Y., Massingham, T., *et al.*: Genomic dna k-mer spectra: models and modalities. Genome Biology **10**(10), 108 (2009)

[16] Zhou, F., Olman, V., Xu, Y.: Barcodes for genomes and applications. BMC Bioinformatics **9**, 546 (2008)

[17] Diaconis, P., Graham, R.L.: Spearman's footrule as a measure of disarray. Journal of the Royal Statistical Society. Series B (Methodological), 262–268 (1977)

[18] Wang, Y., Leung, H.C., Yiu, S.-M., Chin, F.Y.: Metacluster 4.0: a novel binning algorithm for ngs reads and huge number of species. Journal of Computational Biology **19**(2), 241–249 (2012)

[19] Wang, Y., Leung, H.C., Yiu, S.-M., Chin, F.Y.: Metacluster 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample. Bioinformatics **28**(18), 356–362 (2012)

[20] Liao, R., Zhang, R., Guan, J., Zhou, S.: A new unsupervised binning approach for metagenomic sequences based on n-grams and automatic feature weighting. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) **11**(1), 42–54 (2014)

[21] Frigui, H., Nasraoui, O.: Simultaneous clustering and dynamic keyword weighting for text documents. Survey of text mining, 45–72 (2004)

[22] Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. Journal of Machine Learning Research **3**, 993–1022 (2003)

[23] Aso, T., Eguchi, K.: Predicting protein-protein relationships from literature using latent topics. In: Proceedings of The 20th International Conference on Genome Informatics, vol. 23, pp. 3–12 (2009)

[24] Zheng, B., McLean, D.C., Lu, X.: Identifying biological concepts from a protein-related corpus with a probabilistic topic model. BMC Bioinformatics **7**, 58 (2006)

[25] Gerber, G.K., Dowell, R.D., Jaakkola, T.S., Gifford, D.K.: Hierarchical dirichlet process-based models for discovery of cross-species mammalian gene expression. Technical Report (2007)

[26] Chen, X., Hu, X., Lim, T.Y., Shen, X., Park, E., Rosen, G.L.: Exploiting the functional and taxonomic structure of genomic data by probabilistic topic modeling. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) **9**(4), 980–991 (2012)

[27] Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proceedings of the National academy of Sciences of the United States of America **101**(Suppl 1), 5228–5235 (2004)

[28] Pan, X.-Y., Zhang, Y.-N., Shen, H.-B.: Large-scale prediction of human protein- protein interactions from amino acid sequence based on latent topic features. Journal of Proteome Research **9**(10), 4992–5001 (2010)

[29] Shivashankar, S., Srivathsan, S., Ravindran, B., Tendulkar, A.V.: Multi-view methods for protein structure comparison using latent dirichlet allocation. Bioinformatics **27**(13), 61–68 (2011)

[30] Chen, X., Hu, X., Shen, X., Rosen, G.: Probabilistic topic modeling for genomic data interpretation. In: Bioinformatics and Biomedicine (BIBM), 2010 IEEE International Conference On, pp. 149–152 (2010)

[31] Bundschus, M., Dejori, M., Yu, S., Tresp, V., Kriegel, H.-P.: Statistical modeling of medical indexing processes for biomedical knowledge information discovery from text. In: Proceedings of the 8th International Workshop on Data Mining in Bioinformatics (BIOKDD '08), pp. 11–18 (2008)

[32] Yi, Y., Guan, J., Zhou, S.: Effective clustering of microrna sequences by n-grams and feature weighting. In: Proceedings of IEEE 6th International Conference on Systems Biology (ISB'12), pp. 203–210 (2012)

[33] Richter, D.C., Ott, F., Auch, A.F., Schmid, R., Huson, D.H.: Metasim—a sequencing simulator for genomics and metagenomics. PloS One **3**(10), 3373 (2008)

[34] NCBI Acid Mine Drainage Metagenomics Dataset. `http://www.ncbi.nlm.nih.gov/books/NBK6860/`

[35] Bengio, Y.: Learning deep architectures for ai. Foundations and trends® in Machine Learning **2**(1), 1–127 (2009)

[36] Hinton, G., Osindero, S., Teh, Y.-W.: A fast learning algorithm for deep belief nets. Neural computation **18**(7), 1527–1554 (2006)

[37] Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H., *et al.*: Greedy layer-wise training of deep networks. Advances in neural information processing systems **19**, 153 (2007)

[38] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)

[39] Bengio, Y., Courville, A.: Deep learning of representations. In: Handbook on Neural Information Processing, pp. 1–28. Springer, ??? (2013)

[40] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. The Journal of Machine Learning Research **12**, 2493–2537 (2011)

[41] Boulanger-Lewandowski, N., Droppo, J., Seltzer, M., Yu, D.: Phone sequence modeling with recurrent neural networks. In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference On, pp. 5417–5421 (2014). IEEE

[42] Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., Bengio, S.: Why does unsupervised pre-training help deep learning? The Journal of Machine Learning Research **11**, 625–660 (2010)

[43] Alain, G., Bengio, Y., Rifai, S.: Regularized auto-encoders estimate local statistics. Université de Montréal, Tech. Rep. Arxiv report **1211** (2012)

[44] Autoencoder and Sparsity

[45] Denoising Autoencoders. `http://deeplearning.net/tutorial/dA.html#daa`

[46] Yin, C., Yin, X.E., Wang, J.: A novel method for comparative analysis of dna sequences by ramanujan-fourier transform. Journal of Computational Biology **21**(12), 867–879 (2014)

学术论文

发表论文

1. **Ruichang Zhang**, Zhanzhan Cheng, Jihong Guan and Shuigeng Zhou. Exploiting Topic Modeling to Boost Metagenomic Sequences Binning. BMC Bioinformatics, 16(S5): S2, 2015

2. Ruiqi Liao, **Ruichang Zhang**, Jihong Guan, Shuigeng Zhou. A New Unsupervised Binning Approachfor Metagenomic Sequences Based on N-grams and Automatic Feature Weighting. IEEE/ACM Transactions on Compuattational Biology and Bioinformatics, 11(1): 42-54, 2014

3. Hui Liu, **Ruichang Zhang**, Wei Xiong, Jihong Guan, Zhiheng Zhuang, Shuigeng Zhou. A Comparative Evaluation on Prediction Methods of Nucleosome Positioning. Briefings in Bioinformatics, 15: 1014-1027, 2014

致谢

三年的硕士研究生学习即将结束，在这三年里，我的家人，导师，身边的同学和朋友给了我莫大的帮助，在我最困难的时刻给我鼓励和帮助，在我迷茫彷徨时，给我指导和帮助，使我能顺利完成学业，步入社会。在此，我要感谢所有帮助过我的人。

首先，我要感谢我的导师周水庚老师。周老师对我在学术研究中给予了我极大的帮助。周老师总能对一个课题前沿有深刻的见解，对科研工作要求严谨。周老师认真踏实的科研态度，对我的科研和学习生活留下了深刻的影响。同时，每次我遇到科研或者生活中的困难，周老师总能尽力帮助我。衷心感谢周老师这三年多以来的照顾和培养。

其次，我还要感谢实验室这个大家庭的所有同学。跟实验室同学的交流，使我的学生生涯过得更加充实。

最后，我要感谢一直支持我学业的家人，是他们的支持，才能让我克服困难，顺利完成硕士研究生三年的学习。