摘 要

基于 Dirichlet 先验的神经主题模型研究

随着信息技术的发展，互联网每时每刻都在产生海量的文本数据，如何从海量数据中发掘特定的信息是目前机器学习领域亟需解决的难题，主题模型正是解决该问题的关键技术。传统主题模型选用 Dirichlet 分布作为潜在主题的先验分布指导主题词的生成，Dirichlet 分布独特的稀疏性可以帮助模型生成稀疏的主题表示，提高模型的主题提取能力。但传统主题模型复杂的推理算法限制了模型的泛化能力，并且在面对海量的、实时更新的文本数据时，传统主题模型不能快速推理文档主题。

近年来，随着深度学习的发展，将神经网络与主题模型结合进行深层次的主题提取、快速处理大规模数据成为了新的研究热点，这其中以基于变分自编码器（Variational Auto-Encoder，VAE）的主题模型为代表。该类模型要求潜在主题的先验分布具有可重参数化的形式，而传统主题模型中的 Dirichlet 分布因不可重参数化而不能直接作为潜在主题的先验分布应用到神经主题模型中，这使得神经主题模型聚类能力较差。针对这个问题，一些学者提出了将 Dirichlet 先验应用到神经主题模型上的方法，但就现有方法而言，仍存在主题词冗余、重复主题多等问题。因此，为了得到稀疏的主题表示、提高模型的主题提取能力，本文对神经主题模型进行研究，提出了两种更有效的基于 Dirichlet 先验的神经主题模型，具体工作内容如下：

1．使用基于 Kumaraswamy 分布的折棍过程近似 Dirichlet 先验。

1）针对 Dirichlet 分布不能应用在神经主题模型上的问题，提出了一种基于折棍采样的主题模型 KNTM。KNTM 基于神经变分文档模型（Neural Variational Document Model，NVDM）进行改进，用可重参数化的 Kumaraswamy 分布近似不可重参数化的 Beta 分布，然后用以 Kumaraswamy 为基分布的折棍过程近似 Dirichlet 分布。通过这种方式，KNTM 成功将 Dirichlet 分布应用到神经主题模型框架上。

2）在 KNTM 基础上，进一步提出了基于循环折棍构造的 KRNTM 模型。KRNTM 将折棍构造的生成过程与 LSTM 相结合，使用 LSTM 建模长折棍序列，可以动态地为折棍过程分配权重，提高模型的稳定性。

3）通过实验验证模型的有效性，实验结果表明，KNTM/KRNTM 相较于其他神经主题模型能够生成更连贯、质量更高的主题，其中 KRNTM 在提取高维主题时有着更稳定的性能表现。

2．使用基于 Beta 分布的折棍过程近似 Dirichlet 先验。

1)针对 KNTM 在近似 Beta 分布时引入误差的问题，提出了 BNTM 模型。BNTM 直接从 Beta 分布中采样折棍变量构造服从 Dirichlet 先验的潜在主题，帮助模型生成更为稀疏的主题表示，并为了解决 Beta 分布不能重参数化的问题，引入隐式重新参数化的方法来推理参数梯度。相较于 KNTM，BNTM 可以无偏差的估计 Dirichlet 先验。

2）在 BNTM 基础上，进一步提出了基于循环折棍构造的 BRNTM 模型。BRNTM 将从 Beta 分布中采样得到的变量与 LSTM 相结合，更公平的将折棍权重分配给每个主题维度的基分布，这种方法可以有效缓解模型随主题数增多带来的性能下降问题。

3）最后通过实验将 BNTM/BRNTM 和 KNTM/KRNTM 进行对比，结果表明， BNTM/BRNTM 有着更好的主题提取能力，在困惑度、主题一致性和主题唯一性指标上也有着更好的表现。

关键词：

神经网络，主题模型，变分自编码器，Dirichlet 先验，重参数化

Abstract

Research on Neural Subject Model Based on Dirichlet's Prior

With the development of information technology, the Internet generates massive text data all the time. How to discover specific information from massive data is an urgent problem in the field of machine learning. The topic model is the key technology to solve this problem. . Traditional topic models use Dirichlet distribution as the prior distribution of potential topics to guide the generation of topic words. The unique sparsity of Dirichlet distribution can help the model generate sparse topic representations and improve the model's topic extraction ability. However, the complex reasoning algorithm of the traditional topic model limits the generalization ability of the model, and in the face of massive, real-time updated text data, the traditional topic model cannot quickly reason about the topic of the document.

In recent years, with the development of deep learning, it has become a new research hotspot to combine neural network and topic model for in-depth topic extraction and fast processing of large-scale data. Among them, the topic model based on variational autoencoder (VAE) is represented. This type of model requires the prior distribution of latent topics to have a reparameterizable form, and the Dirichlet distribution in traditional topic models cannot be directly applied to neural topic models as the prior distribution of latent topics because it cannot be reparameterized, which makes Neural topic models have poor clustering ability. In response to this problem, some scholars have proposed the method of applying Dirichlet prior to the neural topic model, but as far as the existing methods are concerned, there are still problems such as redundant topic words and many repeated topics. Therefore, in order to obtain sparse topic representation and improve the topic extraction ability of the model, this paper studies the neural topic model and proposes two more effective neural topic models based on Dirichlet prior. The specific work is as follows:

1. Approximate the Dirichlet prior using the zigzag process based on the Kumaraswamy distribution.

1) Aiming at the problem that the Dirichlet distribution cannot be applied to the neural topic model, a topic model KNTM based on zigzag sampling is proposed. KNTM is improved based on the Neural Variational Document Model (NVDM), and the reparameterizable Kumaraswamy distribution is used to approximate the non-reparameterizable Beta distribution, and then the Dirichlet distribution is approximated by the folding stick process based on the Kumaraswamy distribution. In this way, KNTM successfully applies the Dirichlet distribution to the neural topic modeling framework.

2) On the basis of KNTM, a KRNTM model based on cyclic folding stick structure is further proposed. KRNTM combines the generation process of the folding stick structure with LSTM, and uses LSTM to model the long folding stick sequence, which can dynamically assign weights to the folding stick process and improve the stability of the model.

3) The effectiveness of the model is verified by experiments. The experimental results show that KNTM/KRNTM can generate more coherent and higher-quality topics than other neural topic models, and KRNTM has more stable performance when extracting high-dimensional topics.

2. Approximating Dirichlet priors using a broken stick process based on the Beta distribution.

1) Aiming at the problem that KNTM introduces errors when approximating the Beta distribution, a BNTM model is proposed. BNTM directly samples the folded stick variables from the Beta distribution to construct latent topics subject to Dirichlet prior, which helps the model generate more sparse topic representations, and introduces an implicit reparameterization method to solve the problem that the Beta distribution cannot be reparameterized. Inference parameter gradients. Compared with KNTM, BNTM can estimate Dirichlet prior without bias.

2) On the basis of BNTM, a BRNTM model based on cyclic folding stick

structure is further proposed. BRNTM combines the variables sampled from the Beta distribution with LSTM to more fairly assign the weight of the stick to the base distribution of each topic dimension. This method can effectively alleviate the performance degradation of the model as the number of topics increases.

3) Finally, BNTM/BRNTM is compared with KNTM/KRNTM through experiments. The results show that BNTM/BRNTM has better topic extraction ability, and also has better performance in perplexity, topic consistency and topic uniqueness indicators.

Keywords:

Neural Network, Topic Model, Variational Auto-Encoder, Dirichlet prior, Reparameterization

目 录

第 1 章　绪论

1.1　研究背景及意义

近年来随着互联网的普及，人们接收和发布信息的渠道大大拓宽，网络用户可以方便快捷的把信息发布在互联网上，这在给人们带来便利的同时，也造成以文本为代表的网络数据规模呈指数级增长，从互联网海量的文本数据中快速、准确、自动的获取特定信息的需求变得日益迫切。要获取特定的信息，首先需要理解文本所要表达的中心思想，它贯穿文本始终，体现了写作者的真实意图，主题正是这种中心思想的表现形式。

表 1.1　2023 年 1 月 31 日人民日报的版块划分

01 版：要闻	02 版：要闻	03 版：要闻	04 版：要闻
05 版：评论	06 版：要闻	07 版：要闻	08 版：广告
09 版：理论	10 版：经济	11 版：政治	12 版：文化
13 版：社会	14 版：生态	15 版：体育	16 版：国际
17 版：国际副刊	18 版：绿色	19 版：党建	20 版：副刊

图 1.1　环球网（www.huanqiu.com）页面顶端主题索引

如表 1.1 和图 1.1 所示，不同主题的文章被划分到不同的版块下，这种基于主题的内容划分能够让用户高效的发现感兴趣的信息。面对海量的、实时更新的文本数据，对文本进行人工主题分析开销过大，并不现实。因此，如何能实时快速的推理文本主题是目前自然语言处理领域亟待解决的难题，主题模型（Topic Model）正是解决该问题的关键技术。

主题模型是一种用来在文档中发现抽象主题的统计模型，它通过将高维单词空间映射到低维主题空间，能够有效发现文档潜在结构（latent structure）和深层语义信息，用少量的主题词来表示整篇文档的中心思想，最终实现对目标文档的数据降维、语义提取。传统主题模型采用吉布斯采样等推理算法，由于其复杂的推理过程，导致模型的泛化能力较弱，并且不能快速的推理主题，因此无法处理海量的、实时的文本数据。

近年来随着深度神经网络的发展和应用，一些学者开始将传统主题模型与神经网络相结合来提高模型的扩展性和训练速度[1-4]，这其中以基于变分自编码器（VAE）的主题模型为代表。基于 VAE 的主题模型要求先验分布具有重参数化形式，使得梯度能够回传更新参数，相较于传统主题模型泛化能力更强，效率更高。

但神经主题模型同样存在问题。传统主题模型选取 Dirichlet 先验指导主题词的生成，得益于 Dirichlet 分布独特的稀疏性，传统主题模型可以生成稀疏的主题表示，进而聚焦于少量主题，提高模型的主题提取能力。然而，Dirichlet 分布因无法重参数化而不能直接作为潜在主题的先验分布，这使得神经主题模型容易生成冗余的主题词，限制了神经主题模型的应用。因此，从学术和实际应用的角度出发，有必要设计一种能够利用 Dirichlet 先验的神经主题模型，从而生成更具代表性的主题词。

1.2　国内外研究现状

随着信息技术的发展，人们开始对主题建模展开研究，并自主题模型任务提出以来，一直是一个活跃的研究领域。

20 世纪 90 年代 Scott[5]等人提出了基于奇异值分解的 LSA 模型，首次把文本从单词向量空间映射到潜在语义向量空间实现主题的提取。随后，Hofmann[6]提出 PLSA 模型，通过从概率分布中采样的方式代替 LSA 复杂的数学运算，降低了模型的训练成本。Blei[7]在 PLSA 研究的基础上，提出了隐狄利克雷分布模型（Latent Dirichlet Allocation，LDA）。在 LDA 中，每个文档主题的概率分布被赋予了一个 Dirichlet 先验，因为一篇文章的主题往往集中于少数几个单词，Dirichlet 分布优秀的聚类能力恰恰满足主题建模任务的需求。对于生成过程，

LDA 先从基于 Dirichlet 先验的主题分布中为单词选取一个主题，再从基于该主题的单词分布中分配单词，重复此过程最终生成文本。对于参数推理，LDA 常用的推理方法有蒙特卡洛马尔可夫链[8]和变分贝叶斯推理[9-12]，然而前者计算成本过于庞大，后者又缺乏一种通用的参数估计方法，这限制了 LDA 在更大的数据集和更复杂的结构上的扩展性。

近年来神经网络良好的可扩展性和较快的推理速度引起学者的关注，传统主题模型与神经网络的结合成为新的研究热点[20-23]。Miao[13,14]等人首次使用 VAE 结构对文档进行主题建模，提出了 NVDM 模型，该模型将词袋表示的文档作为输入，编码器将输入的文档编码成低维的隐变量并将其作为文档的主题表示，解码器利用隐变量重构出原文档，相较于传统的主题模型，NVDM 训练效率更高、模型的可扩展性更好。Ding[18]等人在 NVDM 的基础上提出了 NTM-R 模型，这种基于正则化约束的方法可以有效的提高主题连贯性。Dieng[19]等人使用单词的词嵌入表示代替传统主题模型的词袋表示，提出了 ETM 模型，该模型将每个单词建模为词嵌入和对应主题嵌入之间的内积，可以获得更多可解释的主题。

需要注意的是，由于 VAE 要求采样过程可重参数化，使得梯度能够反向传播以进行参数训练，以上模型均采用易于重参数化的高斯分布作为潜在主题的先验分布，而没有采用聚类效果更好但不能重参数化的 Dirichlet 分布，这也使得神经主题模型容易生成冗余的主题词，在某些任务中效果不佳[16,23]。为了将 Dirichlet 分布应用于 VAE 框架中，研究人员展开了广泛的研究。Srivastava[15]等人提出了 prodLDA 模型，该模型采用 Laplace 近似的方式用高斯分布近似 Dirichlet 分布，并用得到的 logistic normal 分布作为隐层主题的先验分布，该模型虽然困惑度表现不如 Gaussian VAE，但是得益于 Dirichlet 分布优秀的聚类能力，prodLDA 可以生成更为连贯的主题词，后来 Srivastava 在此工作上进行了扩展，提出了对应的层次结构。由于 Dirichlet 分布可以由 Gamma 分布表示，Joo[16]等人通过求取 Gamma 分布的逆累积分布函数实现了对 Gamma 分布的重参数化，从而间接得到 Dirichlet 分布的重参数化形式。与此类似的是，由于 Weibull 分布与 Gamma 分布具有相似的性质，并且 Weibull 分布具有简易的重参数化形式，Zhang[17] 等人使用 Weibull 分布近似 Gamma 分布，也成功将 Dirichlet 分布应用到 VAE 框架中。Burkhardt[45]等人提出了 DVAE 模型，使用拒绝采样器推断参数

梯度，从而将不可从参数化的 Dirichlet 分布应用到 VAE 框架中。

上述方法可以在一定程度上提高模型对主题词的聚类能力，但仍存在一些不足，例如 prodLDA 不能很好的近似 Dirichlet 分布的多模态特性，DVAE 由于采用拒绝采样的策略，高额的计算复杂度限制了模型的推理速度。所以就现有的研究而言，仍缺少一种方法可以很好的将 Dirichlet 先验应用到神经主题模型中，这也使得神经主题模型生成的主题表示不够稀疏，不能聚焦文档中的少数主题，进而影响模型的主题提取能力。

1.3　研究内容

本文针对基于 VAE 的主题模型中 Dirichlet 分布不能重参数化的问题，将折棍过程[26-29]（Stick-Breaking Processes，SBP）引入神经主题模型，提出了两种基于 Dirichlet 先验的神经主题模型。具体研究内容如下：

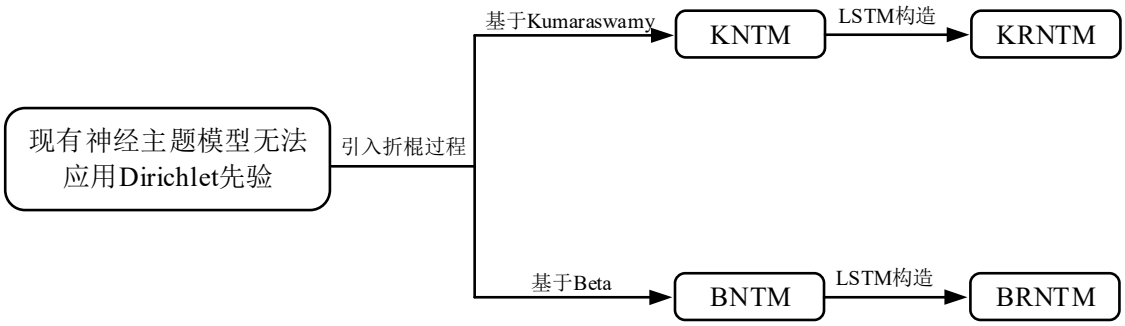


图 1.2　本文研究内容

1．使用基于 Kumaraswamy 分布的折棍过程近似 Dirichlet 先验。

1）为了获得更明确更有意义的主题词，模型选取 Dirichlet 分布作为潜在主题的先验分布，提出了 Kumaraswamy 神经主题模型（Kumaraswamy Neural Topic Model，KNTM）。KNTM 用可重参数化的 Kumaraswamy 分布近似 Beta 分布，并用基于 Kumaraswamy 分布的折棍分布近似 Dirichlet 先验，通过这种方式实现对 Dirichlet 分布的重参数化，成功将 Dirichlet 先验应用到 VAE 框架中。

2）针对 KNTM 在折棍构造过程中表现不够稳定的问题，本文将折棍构造过程与 LSTM 相结合，提出了 Kumaraswamy 循环神经主题模型（Kumaraswamy Recurrent Neural Topic Model，KRNTM）。该模型利用 LSTM 对长折棍序列良好的建模能力，动态地生成折棍权重，提高了模型的稳定性。

3）在主流数据集上验证模型的有效性，结果表明，KNTM/KRNTM 相较于其他神经主题模型能够从语料库中挖掘更连贯主题，其中 KRNTM 能够捕获相邻主题之间的关系，生成一致性更高、性能更稳定的主题词。

2．使用基于 Beta 分布的折棍过程近似 Dirichlet 先验。

1）KNTM 使用可重参数化的 Kumaraswamy 分布近似 Beta 分布，但近似过程会引入误差，且会影响生成变量的概率密度，为了无偏差地估计 Dirichlet 分布，本文直接使用基于 Beta 分布的折棍过程来生成服从 Dirichlet 先验的随机变量，提出了 Beta 神经主题模型（Beta Neural Topic Model，BNTM），并引入隐式重参数化[30]推理采样过程的梯度。

2）在 BNTM 基础上，进一步提出了 Beta 循环神经主题模型（Beta Recurrent Neural Topic Model，BRNTM）。BRNTM 将折棍构造的生成过程与 LSTM 相结合，使用 LSTM 建模长折棍序列，可以动态地为折棍过程分配权重，这种方法可以有效缓解模型随主题数增多带来的性能下降问题。

3）最后通过实验将 BNTM/BRNTM 和 KNTM/KRNTM 进行对比，结果表明， BNTM/BRNTM 在困惑度、主题一致性和主题唯一性指标上也有着更好的表现。

1.4　文章结构

本文共分为五章，具体安排如下：

第一章，绪论。本章首先介绍了本文的研究背景，分析了神经主题模型研究的重要意义。其次概述了国内外神经主题模型的研究现状，尤其是基于 VAE 的主题模型的研究。最后阐述了本文的研究内容并梳理了本文的组织结构。

第二章，相关理论及技术。本章简要介绍了 VAE、折棍过程和循环神经网络的背景知识，包括 VAE 的训练过程、折棍构造的原理形式、长短期记忆网络原理等，为第三、第四章的模型提供理论基础。

第三章，基于 Kumaraswamy 分布的折棍主题模型 KNTM。本章针对 Dirichlet 先验不能应用在神经主题模型上的问题，提出了一种基于折棍采样的主题模型 KNTM。并在 KNTM 的基础上，通过动态地折棍构造提高模型的稳定性，提出了 KRNTM 模型。最后通过实验与其他 Dirichlet VAE 进行对比。

第四章，基于 Beta 分布的折棍主题模型 BNTM。针对 KNTM/KRNTM 近似 Beta 分布过程中引入误差以及影响生成变量概率密度的问题，提出了对应的 BNTM/BRNTM 模型，该模型直接通过以 Beta 为基分布的折棍过程构造 Dirichlet 先验，并通过隐式重参数化的方式推理参数梯度。最后通过实验验证模型的有效性。

第五章，总结与展望。本章总结了本文的主要工作，并对未来的研究方向进行了展望。

第 2 章　相关理论及技术

本章主要介绍本文中所涉及的基础理论知识，包括变分自编码器、折棍过程和长短期记忆网络的背景知识等。

2.1　变分自编码器

变分自编码器（VAE）是一种深度生成模型，由 Kingma[12]等人首次提出，最初用在图像生成任务中，近年来因其对数据潜在分布良好的建模能力备受研究人员的关注。2016 年，Bowman[31]等人开始探索 VAE 在文本领域的应用，利用服从高斯分布的隐变量表示文本语义的潜在分布。此后 VAE 开始逐渐应用在主题模型领域，目前神经主题模型大多基于 VAE 框架。

2.1.1　自编码器

自编码器（Auto-Encoder，AE）是一种无监督的神经网络模型，它可以提取输入样本的潜在特征，并利用潜在特征重构出原始输入样本。自编码器模型主要由编码器（Encoder）和解码器（Decoder）组成，模型结构如图 2.1 所示。编码器能够学习输入样本的潜在特征，将样本压缩至潜在空间表示。解码器能够利用潜在特征重构输入样本。

自编码器可以用于特征提取，与传统的 PCA 等线性方法相比，自编码器凭借神经网络的非线性特征提取能力可以获得更好的数据表示。当给定隐藏层节点数比输入层的维数少的时候，自编码器也可以起到数据压缩的作用。例如，将维度为 1000 的文本数据作为输入，隐含层具有 10 个节点，在损失函数的约束下，自编码器的隐藏层学习到文本中更深层次的语义信息，同时将 1000 维的数据降到 10 维，此时，自编码器也起到了数据压缩的作用。除了进行特征提取、数据降维，自编码器还可以根据学到的潜在特征进行数据生成。

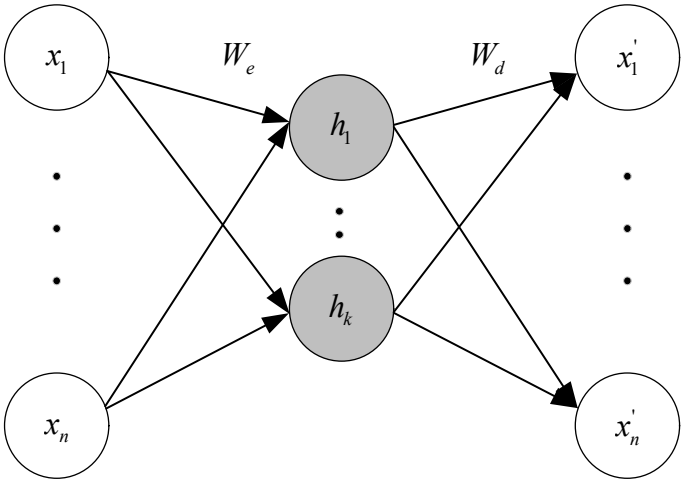


图 2.1　自动编码器的结构

对于输入样本x，自动编码器的编码和解码过程可用如下公式描述：

$$h = \sigma_e(W_e x + b_e) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (2.1)$$

$$x' = \sigma_d(W_d h + b_d) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (2.2)$$

其中W_e、W_d分别为编码器和编码器的权重；b_e、b_d分别为编码器和解码器的偏置；σ_e和σ_d分别为编码器和解码器的激活函数。自编码器的损失函数如下：

$$J(W,b) = \sum_{i=1}^{n} L(x_i, x_i') = \sum_{i=1}^{n} \|x_i' - x_i\|_2^2 \cdots\cdots\cdots\cdots \quad (2.3)$$

自编码器使用反向传播算法（Back Propagation，BP）进行梯度更新。

2.1.2　变分推理

贝叶斯推理是求解不可预测变量后验概率的重要方法，其推理过程开销巨大，但往往很多场景中后验概率并不需要精确求解，于是计算开销更小的变分推理（Variational Inference，VI）算法应运而生[32]。变分推理的核心思想不是精确计算后验分布，而是用一个更简单的分布来近似它，通过变分法将复杂的计算问题转换成参数优化问题，最后将经过迭代后最优的变分分布作为复杂后验分布的代理。目前变分推理广泛应用于计算机领域和数学领域，常被用于处理近似模型复杂的后验分布问题。

在变分推理中，常用 KL 散度[34]（Kullback-Leibler Divergence，KLD）来量化真实后验$p(z \mid x)$与变分近似$q(z)$两分布的近似程度。两个分布差异程度越小，KL 散度就越小，当且仅当两个分布相等时，KL 散度等于零。值得注意的是，KL 散度具有非对称性，$p(z \mid x)$分布到$q(z)$分布的 KL 散度不等于$q(z)$分布到$p(z \mid x)$分布的 KL 散度，即$\mathrm{KL}(p(z \mid x) \| q(z)) \neq \mathrm{KL}(q(z) \| p(z \mid x))$。

当$p(z \mid x)$、$q(z)$为离散变量时，令$P = p_1, p_2, \ldots, p_n$、$Q = q_1, q_2, \ldots, q_n$，其概率分布分别为$p(x)$和$q(x)$，则$P$、$Q$之间的 KL 散度可通过下式得出：

$$D_{KL}(p \| q) = -\sum_x p(x) \log q(x) - \sum_x -p(x) \log p(x)$$

$$= -\sum_x p(x)(\log q(x) - \log p(x))$$

$$= -\sum_x p(x) \log \frac{q(x)}{p(x)} \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \text{（2.4）}$$

由此可得，当$p(z \mid x)$、$q(z)$为连续变量时，KL 散度的定义如下：

$$D(p \| q) = \int_x p(x) \left[\log \left(\frac{p(i)}{q(i)} \right) \right] dx \cdots\cdots\cdots\cdots\cdots \text{（2.5）}$$

此时将近似后验分布问题转化为最小化 KL 散度问题，但由于 KL 散度无法精确计算，导致近似精度不够。针对这一问题，前人已经找到了很好的解决方法：

$$D(q(z) \| p(z \mid x)) = E_q[\log q(z)] - E_q[\log p(z \mid x)]$$

$$= E_q[\log q(z)] - E_q[\log p(x, z)] + \log p(x)$$

$$= \log p(x) - \{E_q[\log p(x, z)] - E_q[\log q(z)]\} \cdots\cdots \text{（2.6）}$$

因为 KL 散度恒大于等于 0，所以公式（2.6）满足以下关系：

$$\log p(x) \geqslant E_q[\log p(x, z)] - E_q[\log q(z)] \cdots\cdots\cdots\cdots \text{（2.7）}$$

不等式左端定义为证据（Evidence），右端则为证据似然$\log(p(x))$的证据下界（Evidence Lower Bound, ELBO），记为$L(q)$，则有：

$$L(q) = E_q[\log p(x, z)] - E_q[\log q(z)] \cdots\cdots\cdots\cdots\cdots \text{（2.8）}$$

此时公式（2.5）KL 散度的最小化可以通过公式（2.7）ELBO 的最大化实现，于是问题转化为了求解最大化 ELBO 的问题，即变分推理需要优化的目标函数

为公式（2.8）。

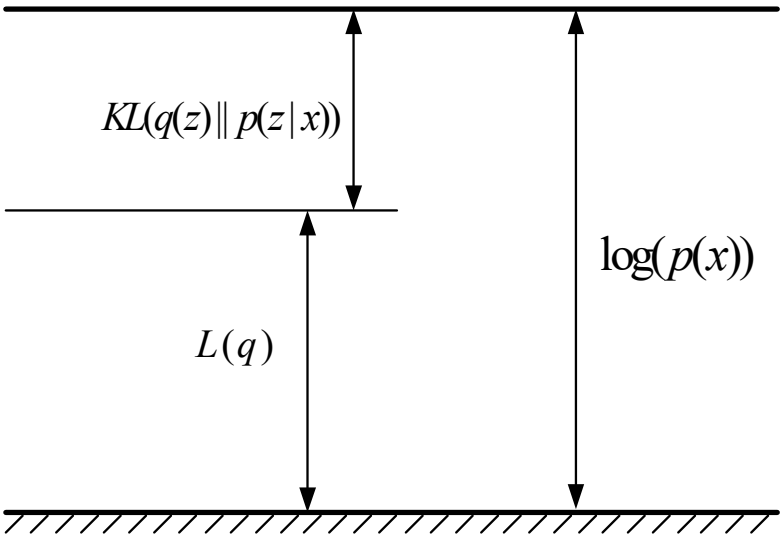


图 2.2　KL 散度与 ELBO 关系示意图

针对变分推理问题的求解有多种方法，其中常用的是基于平均场（Mean-field，MF）理论的方法，该理论认为$q(z)$对隐变量 z 的所有分量是条件独立互不影响的，即满足：

$$q(z) = q(z_1)q(z_2)\cdots q(z_n) \quad (2.9)$$

在这种情况下，每个隐变量z_i可由其自身的变分参数控制，变分分布可因式分解为：

$$q(z) = \prod_{j=1}^{n} q_j(z_j) \quad (2.10)$$

训练过程中采用控制变量的方式简化计算复杂度，通过坐标上升的方法进行迭代求解。

2.1.3　变分自编码器

VAE 在自动编码器的隐层表达上增加一个对隐变量的分布约束，并利用变分推理的算法求解隐变量的变分分布。相较于自动编码器直接通过编码得到隐变量的数值表示，VAE 可以的隐变量从变分分布中采样得到，因此相较于传统的自动编码器，VAE 有着更好的生成能力。

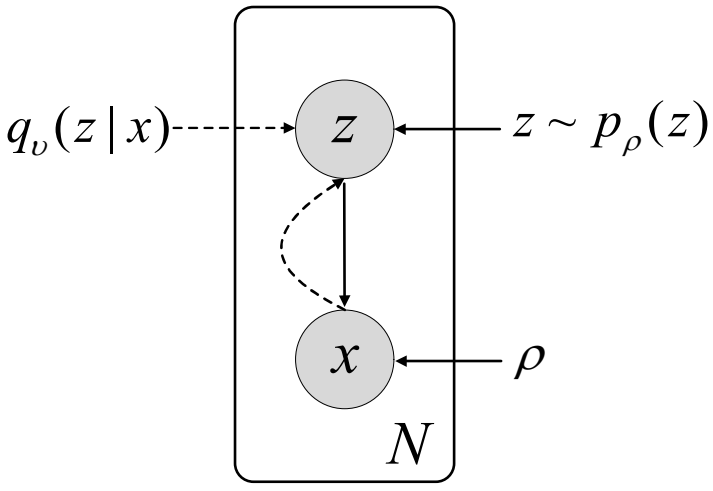


图 2.3　VAE 的概率图模型

如图 2.3 所示，VAE 主要由两部分组成：实线表示解码器$p_\rho(x \mid z)$，虚线表示编码器$q_v(z|x)$，其参数分别为ρ和υ。x表示观测变量，z表示服从先验分布$p(z)$的潜在变量。编码器将观察到的变量x编码为潜在变量z。因为z的真实后验分布往往非常复杂很难计算，隐变量 z 从后验的变分分布$q_v(z|x)$中采样得到，即$z \sim q_v(z \mid x)$，其中参数υ由编码器编码得到。解码器根据潜在变量z重构出x，即$x \sim p_\rho(x \mid z)$。

与 LDA 不同，VAE 采用了自编码器变分贝叶斯（Auto-Encoding Variational Bayes，AEVB）的学习策略。该策略要求隐变量可微且存在非中心参数化（Differentiable Non-Centered Parametrization，DNCP）的表示形式[36]，这样才可以通过蒙特卡洛采样获得期望的近似值。参照公式（2.8）可知，VAE 的优化目标为：

$$log\, p\,(x) \geq \mathcal{L}(x) = \mathbb{E}_{q_{v(z|x)}}\left[log\, p_\rho\,(x \mid z)\right] - \mathrm{KL}\left(q_v(z \mid x) \| p_\rho(z)\right) \cdots （2.11）$$

公式（2.11）的第一项表示从$q_{v(z|x)}$中采样z，使得生成样本中重构出x的几率最大，这部分可以看作模型的重构损失；第二项是使后验分布$q_v(z \mid x)$和先验分布$p_\rho(z)$尽可能的接近，这部分可以看作模型的正则约束，来保证模型的生成能力，防止模型过度拟合塌缩成自动编码器。

模型优化过程依赖于从编码器获得的分布中采样的隐变量z，但采样这一过程是不可导的，这也使得优化参数时梯度不能正常的反向传导，为此需要用到重

参数化技巧（Reparametrisation Trick，RT），即把隐变量用显函数的形式表示出来，通过这种方式巧妙的将随机性转移到变分参数上，将对z的采样变成对变分参数的采样，将对z的求导转换成对变分参数的求导。例如，隐变量z服从高斯分布，则可以写出如下的形式：

$$z = \mu + \delta \cdot \epsilon，\ \epsilon \backsim N(0,1) \cdots\cdots\cdots\cdots\cdots\cdots （2.12）$$

其中μ和δ分别是高斯分布的均值和标准差，ϵ是服从于标准高斯分布的数据噪声。通过这种方式对z采样与直接从高斯分布中采样是一致的，且$\frac{\partial z}{\partial \mu}$和$\frac{\partial z}{\partial \delta}$是连续可导的，通过重参数化使得 VAE 能使用反向传播更新参数。

2.2　折棍采样

折棍采样过程（Stick-breaking Sampling Process，SBSP）是一个随机过程[37]，它可以用于生成服从某个复杂分布的随机变量[26]，也可以用于构造非参统计模型[35]，此外在主题模型领域也有着广泛的应用。在介绍折棍过程之前需要先了解 Beta 分布和 Dirichlet 分布的基础知识。

2.2.1　Beta 分布

Beta 分布描述的是所有概率出现的可能性大小，也称为概率的概率分布。Beta 分布概率密度函数如下：

$$f(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \cdots\cdots\cdots\cdots （2.13）$$

式中α和β的取值决定了 Beta 分布的形态，被称为形状参数，当$\alpha = \beta = 1$时，Beta 分布退化为均匀分布。

在贝叶斯概率理论中，如果后验概率$p(\theta|x)$和先验概率$p(\theta)$满足同样的分布律，则先验分布和后验分布被叫做共轭分布。Beta 分布由二项分布推广得到，并与二项分布互为共轭分布，对于非负实数α和β，有以下关系：

$$Beta(p \mid \alpha,\beta) + BinomCount(m_1,m_2) = Beta(p \mid \alpha + m_1, \beta + m_2) \cdots （2.14）$$

值得一提的是，Beta 分布没有显式的重参数化形式，因此在机器学习中常用 Kumaraswamy 分布作为 Beta 分布的近似分布。Kumaraswamy 分布是具有两个参

数a和b的类 Beta 分布，其取值范围在（0，1）之间。Kumaraswamy 分布具有简易的重参数化形式的同时，与 Beta 分布还有相似的性质[38]。对于任意变量$x \in (0，1)$，其概率密度函数如下：

$$f(x;a,b) = abx^{a-1}(1-x^{a})^{b-1} \cdots\cdots\cdots\cdots\cdots\cdots （2.15）$$

其累积分布函数如下：

$$F(x;a,b) = 1-(1-x^{a})^{b} \cdots\cdots\cdots\cdots\cdots\cdots （2.16）$$

其重参数化形式如下：

$$x\sim\left(1-v^{\frac{1}{b}}\right)^{\frac{1}{a}} \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots （2.17）$$

其中υ ~ Uniform(0, 1)。

2.2.2　Dirichlet 分布

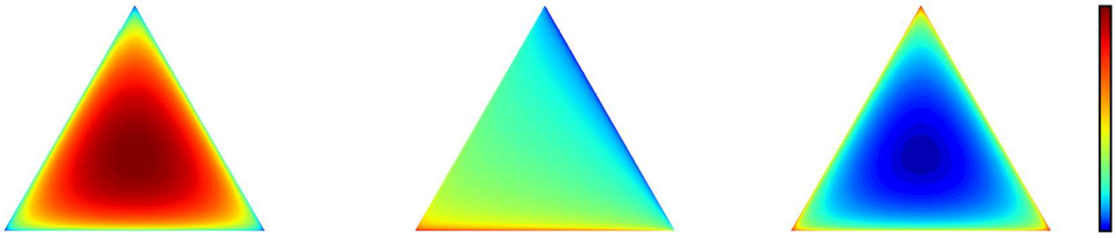
Dirichlet 分布也称作多元 Beta 分布（Multivariate Beta distribution），是 Beta 分布在高维场景下的推广，对于 K 维的 Dirichlet 分布，其概率密度函数如下所示：

$$Dir(\theta \mid \vec{\alpha}) = \frac{\Gamma(\sum_{k=1}^{K}\alpha_k)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\prod_{k=1}^{K}\theta_k^{\alpha_k-1}，\ \sum_{k=1}^{K}\theta_k = 1 \cdots\cdots （2.18）$$

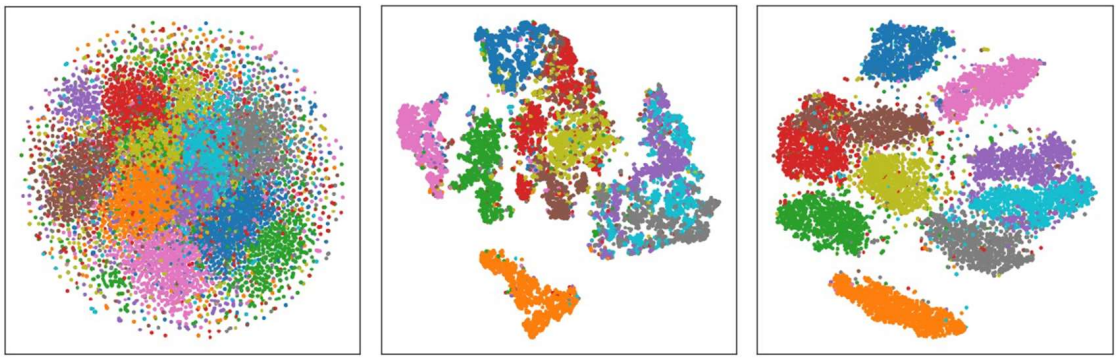
其中$\vec{\alpha} = \{\alpha_1, \alpha_2, \ldots, \alpha_K\}$为 Dirichlet 分布的参数，$\alpha$越大分布越集中，越小越向空间边缘分散。由公式（2.18）知，当 K=2 时 Dirichlet 分布退化为 Beta 分布。与 Beta 分布和二项分布互为共轭分布对应，Dirichlet 分布与多项分布互为共轭分布：

$$Dir(\theta \mid \vec{\alpha}) + MultCount(\vec{m}) = Dir(\theta \mid \vec{\alpha} + \vec{m}) \cdots\cdots （2.19）$$

Dirichlet 分布在主题模型领域扮演着重要角色。由公式（2.19）知，当数据符合多项分布时，参数的先验和后验都能保持 Dirichlet 分布的形式，这条性质为迭代计算带来了便利。因此在主题模型中，当潜在主题以多项分布的形式展现时，采用 Dirichlet 先验在计算上更为便利。



（a）概率单纯形上 Gaussian 分布（左）、GEM 分布（中）和 Dirichlet 分布（右）的密度图，蓝色代表低密度，红色代表高密度



（b）不同分布下基于 t-SNE 的隐变量可视化，Gaussian 分布（左）、GEM 分布（中）和 Dirichlet 分布（右）

图 2.4　不同分布下隐变量密度和分布情况可视化[16]

此外，Dirichlet 分布具有稀疏性。如图 2.4（a）所示，Dirichlet 分布在单纯形的顶点处存在明显的密度峰值，所有密度都分布在概率单纯形的边缘，并且密度与顶点间距很小，这是 Dirichlet 分布所特有的稀疏性。得益于这种特性，以 Dirichlet 分布为先验的主题模型可以生成更为稀疏的主题表示。与此对应的是，Gaussian 分布会给多数变量分配较大的概率密度，不能够聚焦少数主题。

如图 2.4（b）所示，服从 Dirichlet 分布的隐变量具有更好的聚类效果，从而帮助模型从文本中提取更具代表性的主题。而服从 Gaussian 分布的隐变量则较为分散没有明显的类别，生成的主题也是内容宽泛没有重点。

2.2.3　折棍过程

折棍过程的主要思想是将一个单位线段不断分割，每次分割的比例从基分布中采样得到，直到剩余线段的长度趋近于零，或者达到某个停止准则，通过这种方式可以将基分布扩展到更高维度。

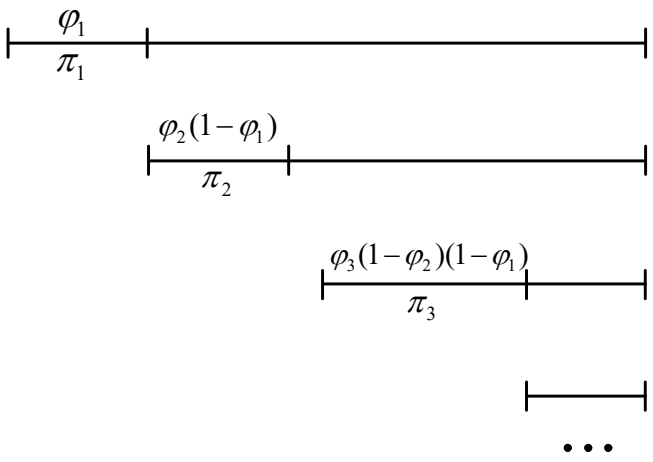


图 2.5　折棍过程示意图

如图 2.5 所示，假设有一根长度为 1 的棍子和一组分布$\phi = \{\varphi_1, \varphi_2, \dots, \varphi_K\}$，对于$\varphi_i \sim p(\pi; \tau_i), \forall \varphi_i \in [0,1]$。第一次从棍子上折取一段比例为$\varphi_1$的棍子，则截取长度$\pi_1 = \varphi_1$；第二次从剩余的棍子上折取一段比例为$\varphi_2$的棍子，则截取长度$\pi_2 = \varphi_2(1-\varphi_1)$，依次循环下去，第$i$次截取长度$\pi_i = \varphi_i(1-\sum_{j=1}^{i-1}\pi_j)$。当总截取次数为$K$时满足下式：

$$\pi_1 + \pi_2 + \cdots + \pi_K = 1 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \text{（2.20）}$$

在上述过程中，$p(\pi; \tau)$称为折棍过程的基分布，τ为基分布的参数。在折棍过程中，选取不同的基分布$p(\pi; \tau)$会使得生成的变量π服从不同的分布。折棍过程的一个应用就是从 Beta 分布中构建服从 Dirichlet 分布的样本，当$p_i(\pi; \tau_i) \equiv Beta\left(x; \alpha_i, \sum_{j=i+1}^{K}\alpha_j\right)$时，经过折棍过程返回的变量$\pi$的概率密度函数如下[33]：

$$p(\pi_1, \dots, \pi_K; \alpha) = \frac{\Gamma(\sum_{i=1}^{K}\alpha_i)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\prod_{i=1}^{K}\pi_i^{\alpha_i - 1} \cdots\cdots\cdots\cdots \text{（2.21）}$$

Dirichlet 分布是顺序可交换的，即给定任何顺序的基分布，从折棍过程返回的随机变量x均可以被置换为$\{x_1, x_2, \dots, x_K\}$，并且不改变其概率密度。这种便利来自于 Beta 分布的对称性：

$$Beta(\alpha, \beta) = \int_1^0 (1-x)^{\alpha-1} x^{\beta-1} d(1-x)$$

$$= \int_0^1 (1-x)^{\alpha-1} x^{\beta-1} dx = Beta(\beta, \alpha) \cdots \text{（2.22）}$$

特别地，设变量x服从以α，β为参数的 Beta 分布，当$\alpha = \beta > 0$时，有$(1-x) \sim$

$Beta(\beta,\ \alpha)$，$p(x)=p(1-x)$。由公式（2.15）可知，Kumaraswamy 分布不存在这种对称性，与之对应地，以 Kumaraswamy 分布为基分布的折棍过程不满足顺序可交换的性质。

2.3　长短期记忆网络

在传统的神经网络中，输入数据和输出数据相互独立，即前时间步的输出不会对本时间步的输出产生影响。但在自然语言处理任务中常常需要记住前面的单词来预测下一个单词或句子，于是循环神经网络应运而生。本节首先介绍循环神经网络的相关知识，然后介绍长短期记忆网络的结构及原理。

2.3.1　循环神经网络

循环神经网络是一种处理时序数据的神经网络，因为其具有很强的学习能力而广泛应用于语音识别、信息抽取等领域，其网络结构如图 2.6 所示。

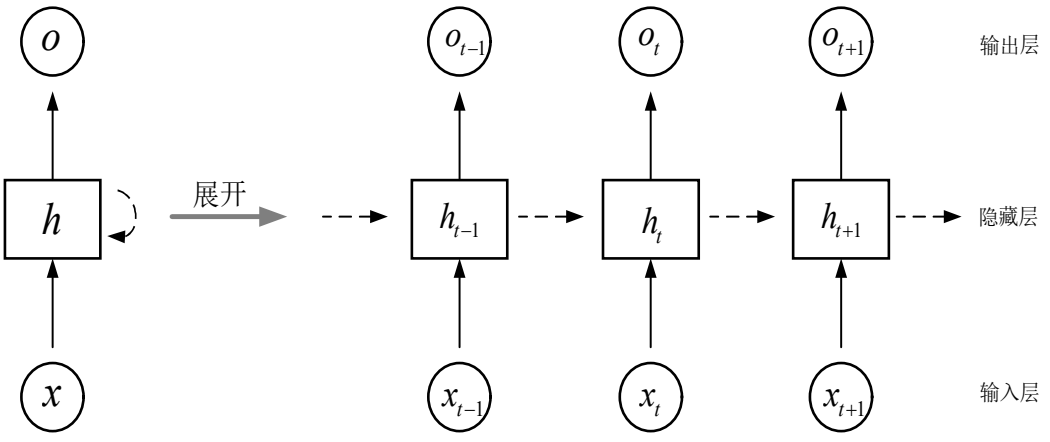


图 2.6　循环神经网络示意图

循环神经网络包含输入层、隐藏层和输出层，其中最关键的结构是隐藏层，它可以记住前文的序列信息。隐藏层中的隐藏状态包含了序列开始到当前时间步的历史信息，由公式（2.23）、公式（2.24）可知，RNN 当前时刻的输出值不仅取决于当前时刻的输入，还与前面多个时刻的状态有关。

$$h_t=\psi(x_tW_1+h_{t-1}W_2+b_1)\cdots\cdots\cdots\cdots\cdots\cdots\ （2.23）$$

$$o_t = h_t W_3 + b_2 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (2.24)$$

$$y_t = \mathrm{softmax}(o_t) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (2.25)$$

其中$\psi(\cdot)$为激活函数，W_1、W_2和W_3为可训练的参数矩阵，b_1和b_2为偏置变量。在循环神经网络中，前一步的输出作为当前步骤的输入，这种串行的网络结构一方面可以保持样本间的依赖关系，另一方面可以通过参数共享大大减少所需训练参数的数量。时间序列中往往包含了大量的上下文信息，得益于独特的内循环网络结构，循环神经网络能够充分利用序列中的语义信息，这也使得循环神经网络具有良好的序列处理及语言建模能力。循环神经网络是深度神经网络的一种，其参数训练过程采用随时间反向传播算法，在处理较长序列时，往往会产生梯度爆炸（Gradient Exploding，GE）或梯度消失（Gradient Vanishing，GV）的问题。

2.3.2　长短期记忆网络

在实际应用中，循环神经网络往往会因为学习路径过长导致梯度消失，即模型无法从前向序列中获得有效的信息。针对这个问题，人们提出了长短期记忆网络[39](Long Short Term Memory，LSTM)来解决循环神经网络短期记忆的瓶颈。

LSTM 基于循环神经网络构建，相较于循环神经网络使用单条路径传递信息，LSTM 新增了单元状态（Cell State，CS）保存长期依赖的信息。单元状态只参与少量的线性交互，这样可以保证 LSTM 可以传递长期信息且很少衰减。单元状态类似于神经网络的一条传送带，将长期信息传递到当前时刻，保证了网络的长期记忆能力。此外，LSTM 引入了门函数的概念，它通过门函数控制着信息的通过量。在 LSTM 中存在三种门，分别是决定是否从单元状态中丢弃信息的遗忘门（Forget Gate，FG）、决定新信息是否存放在单元状态的输入门（Input Gate，IG）与根据单元状态确定输出值的输出门（Output Gate，OG）。长短期记忆网络的整体结构如图 2.7 所示：

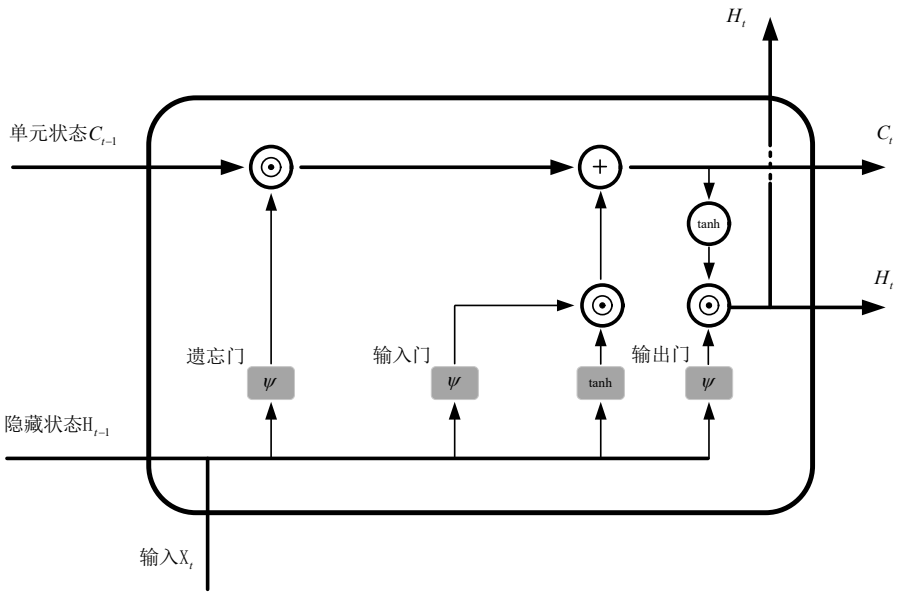


图 2.7　长短期记忆网络示意图

LSTM 的第一步是决定是否要从前文序列中丢弃信息，这个决定由遗忘门完成。遗忘门首先读取上一个时刻的隐藏状态H_{t-1}和输入x，输出一个介于 0 和 1 之间的数值传递给上一个时刻的单元状态C_{t-1}，1 表示全部保留上时刻单元状态C_{t-1}，0 表示全部舍弃上时刻单元状态C_{t-1}，输出F_t如下所示：

$$F_t = \psi(X_t W_1 + H_{t-1} U_1 + b_1) \cdots\cdots\cdots\cdots\cdots\cdots （2.26）$$

其中$\psi(\cdot)$表示激活函数，W_1和U_1表示可训练的矩阵参数，b_1表示偏置变量。遗忘门是长短期记忆网络的关键，通过控制遗忘门可以避免因为学习路径过长导致梯度消失或梯度爆炸的问题。

LSTM 的第二步是决定多少信息加入到单元状态。首先通过输入门决定哪些信息更新，其次 tanh 层生成候选记忆$\overline{C_t}$,最后将输入门的输出I_t和候选记忆$\overline{C_t}$结合对单元状态更新：

$$I_t = \psi(X_t W_2 + H_{t-1} U_2 + b_2) \cdots\cdots\cdots\cdots\cdots\cdots （2.27）$$

$$\overline{C_t} = \tanh(X_t W_3 + H_{t-1} U_3 + b_3) \cdots\cdots\cdots\cdots\cdots\cdots （2.28）$$

$$C_t = F_t \odot C_{t-1} + I_t \odot \overline{C_t} \cdots\cdots\cdots\cdots\cdots\cdots （2.29）$$

其中$\psi(\cdot)$表示激活函数，W_2、W_3、U_2和U_3表示可训练的矩阵参数，b_2、b_3表示偏置变量。

LSTM 的最后一步是输出当前时刻隐藏状态H_t的值，这个输出的值会基于单

元状态C_t得到：

$$O_t = \psi(X_t W_4 + H_{t-1} W_5) \cdots\cdots\cdots\cdots\cdots\cdots \quad (2.30)$$

$$H_t = O_t \odot \tanh(C_t) \cdots\cdots\cdots\cdots\cdots\cdots \quad (2.31)$$

LSTM 解决了序列长期依赖的问题，此后也出现了对 LSTM 的改进工作。例如 Cho 等人[40]提出了结构更为简单的门控循环单元（GRU），通过定义的重置门更新们简化 LSTM 复杂的计算过程；Lei 等人[41]为了加速网络的训练提出了简单循环单元（SRU），SRU 不依赖之前时间步的完整计算，从而实现并行化计算；双向长短期记忆网络[42]（BiLSTM）将两个方向相反的 LSTM 叠加在一起，对于每个时刻同时提供前后两个方向的信息，相较于 LSTM 可以更好的捕获相邻词语之间的关系。

2.4　本章小节

本章介绍了本文所涉及的相关理论及技术，总共分为三个部分。首先系统地阐述了变分推理的原理以及 VAE 的训练过程。其次介绍了 Beta 分布、Kumaraswamy 分布和 Dirichlet 分布的性质，并描述了折棍构造的过程及基本性质。最后介绍了循环神经网络的原理，以及长短期记忆网络的结构、原理等。本章为第三章和第四章的模型提供了理论基础。

第 3 章　基于 Kumaraswamy 分布的折棍主题模型 KNTM

基于变分自编码器（VAE）的神经主题模型要求潜在主题的先验分布具有可重参数化的形式，而传统主题模型中 Dirichlet 先验虽然具有稀疏性可以提高模型对主题词的聚类能力，但是因无法重参数化不能将其应用到神经主题模型中。本章针对该问题提出了一种基于折棍构造的主题模型 KNTM，该模型可以利用 Dirichlet 先验获得稀疏的主题表示。为了提高 KNTM 的稳定性，本章进一步提出了基于循环折棍构造的主题模型 KRNTM。本章的结构如下：首先给出了 KNTM 的推理算法和模型结构；其次介绍了循环折棍构造的过程以及 KRNTM 的推理算法；最后通过在主流数据集上的实验验证两个模型的有效性。

3.1　KNTM 模型

为了解决 Dirichlet 先验不能直接应用在基于 VAE 的神经主题模型上的问题，本章提出了一种新的主题模型——Kumaraswamy 神经主题模型（KNTM）。由 2.2.3 可知，当任意基分布服从$p_i(\pi;\varphi_i) \equiv Beta\left(x;\alpha_i,\sum_{j=i+1}^{K}\alpha_j\right)$时，由折棍过程构造的随机变量服从 Dirichlet 分布[33]。借助于这条性质，KNTM 在 NVDM 框架上进行改进，用可重参数化的 Kumaraswamy 分布代替不可重参数化的 Beta 分布，并用以 Kumaraswamy 为基分布的折棍过程构造 Dirichlet 分布，从而将 Dirichlet 先验应用到基于 VAE 的神经主题模型上。通过这种方式，KNTM 可以利用 Dirichlet 分布的稀疏特性，帮助模型生成更为稀疏的主题表示。接下来，本节将从生成过程、主题推理和 VAE 结构三个方面介绍 KNTM 模型。

3.1.1　生成过程

给定包含D个文档的语料库，每篇文档x_{d}包含的单词数为N_d，KNTM 从以$Kumaraswamy(a,b)$为基分布的折棍过程构造主题分布θ_d，之后从θ_d中采样一个主题z_{dn}，并从该主题的单词分布中采样得到单词x_{dn}，重复此过程N_d次即可生成文档。综上，对于每篇文档x_{d}，KNTM 模型的生成过程如下所示：

① 采样原始的主题分布$\lambda_d \sim f_{SB}(Kumaraswamy(a,b))$

② 计算真实的主题分布$\theta_d = \text{softmax}(W^T\lambda_d)$

③ 对每个单词$x_{dn}, n \in [N_d]$

- 采样一个主题$z_{dn} \sim \text{Multinomial}(\theta_d)$
- 生成单词$x_{dn} \sim \text{Multinomial}(\beta_{z_{dn}})$

其中$f_{SB}(\cdot)$表示折棍过程；Multinomial(·)表示多项式分布。

3.1.2　主题推理

本文主题推理的目的是通过拟合给定文本数据$x \in \mathbb{R}^V$，找到文档潜在的主题分布$\theta \in \mathbb{R}^K$。给定一组服从 Kumaraswamy 分布的向量$\phi = \{\varphi_1, \varphi_2, \dots, \varphi_{K-1}\}$，任意 $i \in [1, K-1]$，$\varphi_i \sim Kumaraswamy(\varphi; a_i, b_i)$，参数 $(a,b) = \{a_1, a_2, \dots, a_{K-1}; b_1, b_2, \dots b_{K-1}\}$由神经网络$f_\rho(x)$得到，其中$\rho$为神经网络的参数。因此，$\phi$可以表示为如下形式：

$$\phi \sim Kumaraswamy(a,b),\ (a,b) = f_\rho(\phi \mid x) \quad \cdots\cdots（3.1）$$

未转换的主题分布λ可由以ϕ为基分布的折棍过程生成。由 2.2.3 可知，折棍过程生成的是无限维度的概率向量，此时隐变量是非参数的，当它不能充分表示数据潜在特征时，其维度会增加，计算也会变的十分复杂，且非参模型在处理大规模数据集时，表现往往不佳。因此本文采取截断的方式对文档中的有限主题进行建模，即：

$$\lambda_k = \begin{cases} \varphi_1, k = 1 \\ \varphi_k \prod_{i<k} (1-\varphi_j), 1 < k < K \\ 1 - \sum_{i=1}^{K-1} \lambda_i, k = K \end{cases} \quad \cdots\cdots\cdots\cdots（3.2）$$

得到未转换的主题分布λ后，再经过一个softmax层得到最终的文档-主题分布θ：

$$\theta = \text{softmax}(W^T\lambda) \quad \cdots\cdots\cdots\cdots\cdots\cdots（3.3）$$

其中$W \in \mathbb{R}^{K \times K}$表示可训练的线性转换矩阵。

给定文档集$\{x_1, x_2, \ldots, x_D\}$，其中$x_{\mathrm{d}}$包含$N_{\mathrm{d}}$个单词，则文档集的边际似然函数如下所示：

$$\mathcal{L}(\rho, v) = logp(x) = \sum_{d=1}^{D} log[p(x_d \mid \rho, v)]$$

$$= \sum_{d=1}^{D} \int p(\lambda_d) \prod_{n=1}^{N_d} p(x_{dn} \mid \lambda_d, \rho, v) d\lambda_d \cdots （3.4）$$

其中λ_d表示第d篇文档原始的主题分布，假设单词分配的主题为z_{dn}，则该词的条件概率$p(x_{dn} \mid \lambda_d, \rho, v)$如下所示：

$$p(x_{dn} \mid \lambda_d, \rho, v) = \sum_{k=1}^{K} \theta_{dk} \beta_{k, x_{dn}} \cdots\cdots\cdots\cdots （3.5）$$

其中θ_d表示第d篇文档真实的主题分布，$\beta_{k, w_{dn}}$表示服从第k个主题的单词分布。

综上，对于每篇文档x_{d}，模型的损失函数如下所示：

$$\mathcal{L}_d(\theta, \rho, v) = \sum_{n=1}^{N_d} \mathbb{E}_q \, log\left[p\left(x_n \mid \beta, \hat{\theta}\right)\right] - D_{KL}\left(MVK(a, b) \| Dirichlet(\vec{\alpha})\right)$$

$$\cdots\cdots （3.6）$$

其中$\vec{\alpha}$是 Dirichlet 分布的超参数。对于等式（3.6）的第一项，本文采取与其他 Dirichlet VAE 模型相同的随机梯度变分贝叶斯算法（Stochastic Gradient Variational Bayes，SGVB）来计算该期望值：

$$\mathbb{E}_{q_\rho\left(\phi \mid x_{1:N_d}\right)}\left[log\, p\left(x_{1:N_d} \mid \beta, \hat{\theta}\right)\right] = \frac{1}{S} \sum_{s=1}^{S} \sum_{n=1}^{N_d} logp(x_n \mid \theta^{(s)}, \beta) \cdots （3.7）$$

其中S表示蒙特卡洛采样的个数。

对于等式（3.6）的第二项，将 $D_{KL}(MV\mathrm{K}(x; a, b) \| \, Dirichlet(\alpha))$ 分解为在相应主题维度上 Kumaraswamy 分布与 Beta 分布 KL 散度的和：

$$\sum_{i=1}^{K-1} D_{KL}\left(Kumaraswamy\left(f_\rho(a_i|x), \sum_{j=i+1}^{K} f_\rho\left(a_j|x\right)\right) \| \, Beta\left(\alpha_i, \sum_{j=i+1}^{K} \alpha_j\right)\right)$$

$$\cdots\cdots （3.8）$$

在每一个主题维度$k \in \mathbb{R}^{K-1}$上，Kumaraswamy 分布与 Beta 分布的 KL 散度可表示成如下形式：

$$D_{KL}(Kumaraswamy(a_k,b_k)\|Beta(\mu_k;\eta_k))$$
$$=\frac{a_k-\mu}{a_k}\left(-\gamma-\Psi(b_k)-\frac{1}{b_k}\right)+log(a_kb_k)+log\,B(\mu_k,\eta_k)$$
$$-\frac{b_k-1}{b_k}+(\eta_k-1)b_k\sum_{m=1}^{\infty}\frac{1}{m+a_kb_k}B\left(\frac{m}{a_k},b_k\right)\cdots\cdots\cdots\quad(3.9)$$

其中a_k、b_k和μ_k、η_k分别是 Kumaraswamy 分布和 Beta 分布的参数，γ是欧拉常数，$\Psi(\cdot)$代表 Digamma 函数，$B(\cdot)$代表 Beta 函数。

KNTM 完整的主题推理过程如算法 3.1 所示。

算法 3.1　Approximating inference for KNTM.

Initialize variational neural network parameter ρ and v
Given hyperparameter $\{\vec{\alpha},K\}$
While epoch < EPOCH do
　Draw an unnormalized topic proportion
　　Compute the parameter a and b of the distribution
　　Sample: $\varphi_1\sim p_1(\varphi;a_1,b_1)$,where p denote Kumaraswamy distributions
　　Assign: $\lambda_{d1}=\varphi_1,i=2$
　　While i<K do:
　　　Sample: $\varphi_i\sim p_i(\varphi;a_i,b_i)$
　　　Assign: $\lambda_{di}=\varphi_i\prod_{j<j}(1-\varphi_j)$
　　End While
　　$\lambda_{dK}=1-\sum_{i=1}^{K-1}\lambda_{di}$
　Compute the topic proportion $\theta_d=\mathrm{softmax}(W^T\lambda_d)$
　For each word token $x_{dn},n\in[N_d]$
　　Compute a word $p(x_{dn}|\theta_d)=\mathrm{softmax}(\theta_d^T\beta_{z_{dn}})$
　End For
　　Update neural network parameter with their gradients
　　epoch = epoch + 1
End While

3.1.3　VAE 结构

与其他 Dirichlet VAE 一样，KNTM 遵循 NVDM 框架，因此可以从 VAE 的角度解释 KNTM。KNTM 将变分分布$q(\theta_{1:D}|x)$作为编码器，从文本中推理潜在的文本-主题分布$\theta_{1:D}$，将$p\,(x\mid\theta_{1:D},\beta)$作为解码器建模文档的生成过程。模型

的整体结构如图 3.1 所示：

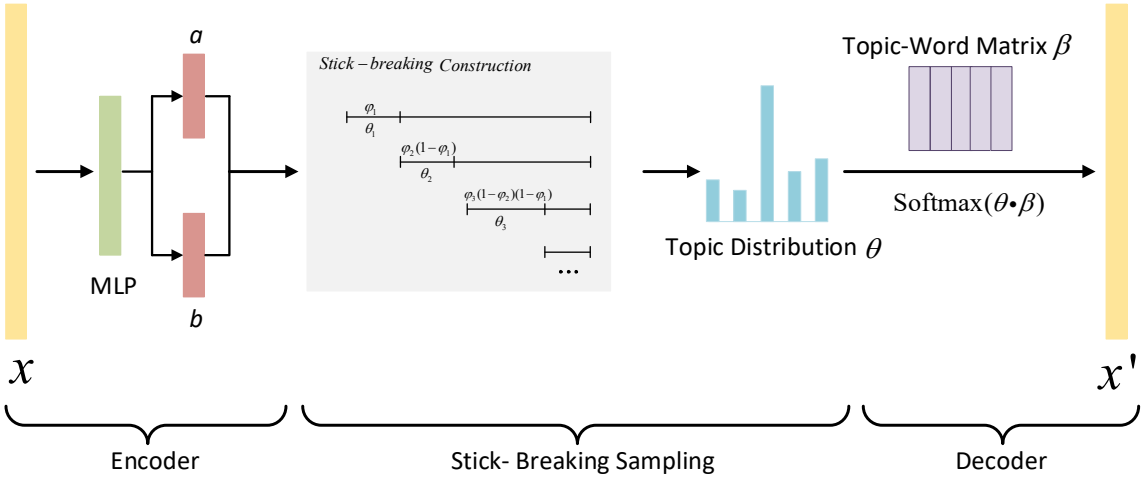


图 3.1　KNTM 结构示意图

编码器 将文档-主题的先验分布设置为 Dirichlet 分布，这是多项式分布的共轭先验分布。由于 Dirichlet 分布不能重参数化，不能直接将其作为先验分布用于 VAE 框架，为此本文没有直接从 Dirichlet 分布中对文档-主题分布λ进行采样。本文使用以 Kumaraswamy 为基分布的折棍分布近似 Dirichlet 先验，并在折棍构建过程中完成对λ的采样，此时λ服从多元 Kumaraswamy 分布（Multivariate Kumaraswamy，MVK），即：

$$\lambda \sim MVK(x; a, b) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (3.10)$$

其中a, b为 MVK 分布的参数。由公式（2.21）可知，λ表现出与传统 Dirichlet 先验相似的主题分布。本文引入一个推理网络来构建 Kumaraswamy 分布，输入文档的词表示$x \in \mathbb{R}^V$，参数a通过多层感知器（Multilayer Perceptron，MLP）获得，参数b在a的基础上得到：

$$a = \psi(Wx + c) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (3.11)$$

$$b_i = \sum_{j=i+1}^{K} a_j \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (3.12)$$

其中$\psi(\cdot)$表示 softplus 激活函数；W是可训练的矩阵参数；c是网络的偏置参数。

解码器 将原始的文档-主题分布λ进行线性变换，得到最终的主题分布θ，再使用对数 softmax 变换来重构出文本x。假设一篇文档的词汇量为 V，将解码器中的权重矩阵视为主题和单词之间的分布关系β，解码器使用以下方法利用θ重构出V维