基于潜在狄利克雷分配模型的语义搜索方法

张桐[1]

1.中国地质大学（武汉） 计算机学院，武汉 430078

摘 要：传统的基于关键字的文本信息检索方法缺少对文本语义的搜索能力。本文基于 Latent Dirichlet allocation（潜在狄利克雷分配）模型，使用 Gibbs 抽样方法对文本数据进行建模，得到文本的主题分布，并以此实现文本数据的语义搜索。实验使用短文本查询，对不同相似度度量、分词策略和主题数下的模型效果进行了对比和分析。最终结果表明，该方法能够达到较高的搜索准确度。

关键词：信息检索,主题模型,LDA,Gibbs 抽样

中图分类号：TP301 **文献标志码：**A

引用格式：张桐.2022. 基于潜在狄利克雷分配模型的语义搜索方法.

1 引 言

IDC 研究估计，在 2025 年之前，企业数据将以 40~50% 的复合年增长率增长，每两到三年翻一倍。增长的数据中 80~90% 是与非结构化数据相关的。非结构化数据越来越多地产生于社交媒体、搜索引擎查询、实时流媒体和物联网传感器。(Goodwin 2019) 文本数据在非结构化数据中占有重要地位，对文本数据的处理与分析正变得越来越重要。

传统的文本信息检索方法是关键字搜索，例如经典的倒排索引（inverted index）方法，缺乏对文本语义的搜索能力,存在一定的局限性。本文使用 LDA 模型对文本数据的潜在语义结构进行主题建模，实现了一种基于 LDA 模型的语义搜索方法。

2 研究方法或原理

2.1 Latent Dirichlet Allocation 模型

Latent Dirichlet Allocation（潜在狄利克雷分配，LDA）是由 (Blei 2003) 提出的一种用于离散数据集合的概率生成模型。LDA 模型常被用于在自然语言处理中对语料库进行建模，进而实现文档聚类、主题挖掘、情感分析、推荐系统等功能；LDA 模型在图像处理和计算机视觉领域也有应用。

LDA 模型的基本假设是：一篇文档是由若干个主题混合而成的，每个主题是由若干的词语构成的。LDA 采用词袋模型，将每篇文档看作词语的集合，而忽略词语之间的先后位置关系。

LDA 模型的具体生成过程如下图 (Blei 2003) 所示：

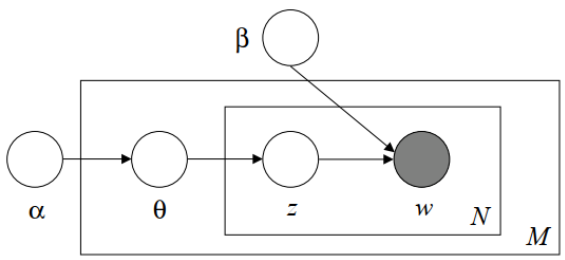

图 1 LDA 模型的图形表示。方盒代表重复过程，外侧的方盒代表文档的重复生成，内侧的方盒代表主题和词语的重复选取。

Fig. 1 Graphical model representation of LDA. The boxes are “plates” representing replicates. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

在生成文档时，LDA 模型首先会从一个主题分布中生成 K 个主题，并进行编号。之后，从文档-主题编号分布的分布中生成一个文档-主题编号分布，进而生成一个主题编号，并挑选编号相符的主题，生成一个词语。重复该过程，直到达到要求的词语数。其中主题分布是以β为参数的 Dirichlet 分布，主题是以ϕ_z为参数的 Multinomial 分布，而文档-主题编号分布的分布是一个以α为参数的 Dirichlet 分布，文档-主题编号分布是一个θ为参数的 Multinomial 分布。

根据以上定义可以得到 $\theta,\vec{z},\vec{w}$ 的联合分布

$$p(\theta,\vec{z},\vec{w}|\alpha,\beta)=p(\theta|\alpha)\prod_{n=1}^{N}p(z_n|\theta)p(w_n|z_n,\beta)$$

对 θ 积分可得 $\vec{w},\vec{z}$ 的联合分布：

$$p(\vec{w},\vec{z}|\alpha,\beta)=\int p(\theta|\alpha)\prod_{n=1}^{N}p(z_n|\theta)p(w_n|z_n,\beta))\,d\theta$$

对 θ 进行积分，并对$\vec{z}$进行求和，可以得到 $\vec{w}$ 的分布：

收稿日期：2022-12-23

第一作者简介：张桐，2002 年生,男,本科生。E-mail: ztong03@hotmail.com

$$p(\vec{w}|\alpha,\beta) = \int p(\theta|\alpha)\left(\prod_{n=1}^{N}\sum_{z_n} p(z_n|\theta)p(w_n|z_n,\beta)\right)d\theta$$

2.2 Gibbs 抽样

LDA 模型参数的推断方法有变分贝叶斯（variational Bayes）、最大似然（likelihood maximization）和 Gibbs 抽样（Gibbs sampling）。其中 Gibbs 抽样是目前最为流行的方法。

根据 $\vec{w},\vec{z}$ 的联合分布可以得到相应的 Gibbs 抽样公式(Griffiths and Steyvers 2004)：

$$p(z_i = k|\vec{z}_{\neg i},\vec{w}) \propto \frac{n_{m,\neg i}^{(k)} + \alpha_k}{\sum_{k=1}^{K}\left(n_{m,\neg i}^{(k)} + \alpha_k\right)} \cdot \frac{\left(n_{k,\neg i}^{(t)} + \beta_t\right)}{\sum_{t=1}^{V}\left(n_{k,\neg i}^{(t)} + \beta_t\right)}$$

其中 z_i 表示第 i 个词的主题，$\vec{z}_{\neg i}$ 表示其它词语的主题，$n_m^{(k)}$ 表示 m 文档中出现 k 主题的次数，$n_k^{(t)}$ 表示 k 主题中出现 t 词的次数。

2.3 相似度度量

在通过 Gibbs 抽样推断 LDA 模型参数，进而推断出文档的主题分布后，可以使用以下相似度度量来评价两个主题分布θ_i和θ_j间的距离：

（1）余弦相似度：

$$S_C(\theta_i,\theta_j) = \frac{\theta_i \cdot \theta_j}{\|\theta_i\|\|\theta_j\|}$$

（2）Kullback-Leibler 散度（相对熵）(Lin 1991)：

$$D_{KL}(P||Q) = \sum_{x\in X} P(x)\log\frac{P(x)}{Q(x)}$$

其中P和Q是两个离散概率分布，X为它们的样本空间，$P(x)$和$Q(x)$为相应的概率分布函数。

当进行单次观察时，多项分布会退化为分类分布，此时相应的 KL 散度为：

$$D_{KL}(\theta_i||\theta_j) = \sum_{k=1}^{n}\theta_{ik}\log\frac{\theta_{ik}}{\theta_{jk}}$$

由于 KL 散度不是对称的，不便于用于评价相似度，一般会使用它的变体 Jensen-Shannon 散度(Lin 1991)：

$$D_{JS}(P||Q) = \frac{1}{2}D_{KL}(P||M) + \frac{1}{2}D_{KL}(Q||M)$$

其中$M = \frac{1}{2}(P+Q)$，是P和Q两个分布的平均。

对于分类分布来说，JS散度为：

$$D_{JS}(\theta_i||\theta_j) = \frac{1}{2}D_{KL}(\theta_i||\frac{\theta_i+\theta_j}{2}) + \frac{1}{2}D_{KL}(\theta_j||\frac{\theta_i+\theta_j}{2})$$

（3）除了以上使用主题分布来度量文档相似度的方法外，我们也可以使用一篇文档的主题分布生成另一篇文档的概率来作为相似度的度量：

$$S_p = \prod_{w\in W_i}\sum_{k=1}^{n} p(w|z_k)p(z_k|W_j)$$

其中W_i是被生成的文档，W_j则是另一篇文档。

3 数据结果处理与分析

3.1 语料库及预处理

本次实验使用中文维基百科的词条数据作为语料库(“Zhwiki Dump Progress on 20221201” 2022)。截止 2022 年 12 月 1 日，中文维基百科共有 8197491 个词条。其中前 69134 个词条的长度直方图如下：

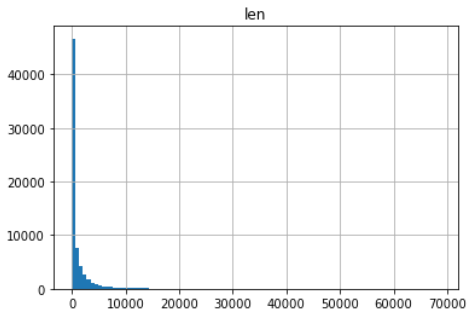

图 2 前 69134 个词条的长度直方图

Fig. 2 Length histogram of the first 69,134 pages

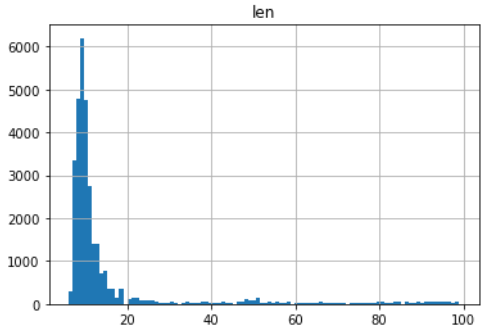

图 3 前 69134 个词条中长度小于 100 字符的词条的长度直方图

Fig. 3 Histogram of the length of the first 69,134 pages with less than 100 characters

通过直方图可以发现，维基百科中有含有大量短词条，约 45%的词条长度小于 100 字符。由于 LDA 模型对于短文本的建模效果不佳，是否过滤短词条可能会对模型效果产生影响。

为了便于进行实验，我们对词条数据进行如下预处理：

1. 语料切分：选取前 1000 个词条。
2. XML 格式转换：词条的原始 XML 数据包含着许多对于本实验没有帮助的元数据，对 XML 进行格式转换能够简化数据读取。

3. 简繁转换：许多词条是使用繁体中文编写的，不利于 LDA 模型的训练，因此我们使用 OpenCC (Kuo [2010] 2010) 将词条中的繁体中文转换为简体中文。
4. 分词：使用 jieba (Sun 2012)，通过前缀词典和 HMM 模型对词条文本进行词图扫描，基于词频进行切分，实现对词条文本的分词。

3.2 模型训练及评价方法

我们使用 Genism (Řehůřek and Sojka [2011] 2010) 完成对 LDA 模型的训练。定义搜索准确度为使用词条标题进行搜索时，被对应词条的排名超越的词条比例。以所有词条的平均准确度作为模型性能的评价依据，平均准确度越高，认为模型的搜索效果越好。

3.3 主题数

在不同的文本数据和相似度度量下，使用不同的主题数训练 LDA 模型，比较模型性能：

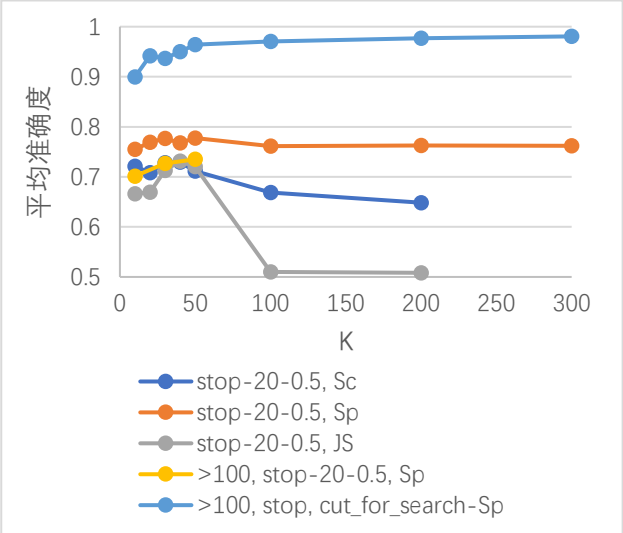

图 4 不同的文本数据、相似度度量和主题数 K 下的平均准确度

Fig. 4 Average accuracy with different text data and similarity measures and number of topics K

实验结果表明，主题数会对语义搜索的准确度产生影响。在不同的文本数据和相似度度量下，最佳的主题数也会有所不同，需要通过试验进行选取。

3.4 相似度度量

在不同的相似度度量下，使用不同的主题数训练 LDA 模型，比较模型性能：

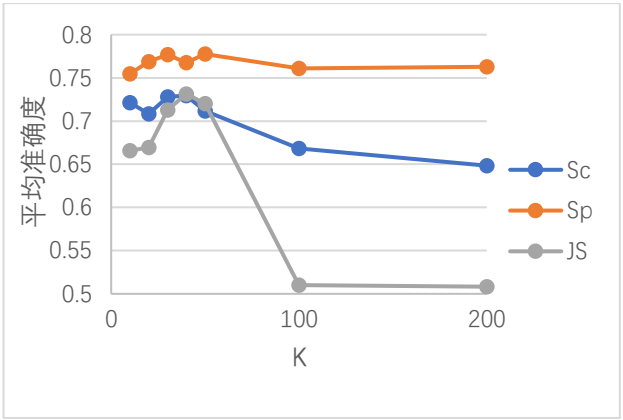

图 5 不同相似度度量和主题数下的平均准确度

Fig. 5 Average accuracy with different similarity measures and number of topics

实验结果表明，在用于进行语义搜索时，生成概率的效果好于余弦相似度，余弦相似度的效果又好于 Jensen-Shannon 散度。之所以会出现这种现象，是因为对于短文本查询来说，推断出的主题分布会丢失部分语义，导致匹配效果不佳，而使用生成概率作为相似度度量避免了对短文本进行主题分布推断，保留了原始语义。

3.5 分词策略

在生成概率度量下，使用不同的词频过滤条件和分词方法训练 LDA 模型，比较模型性能：

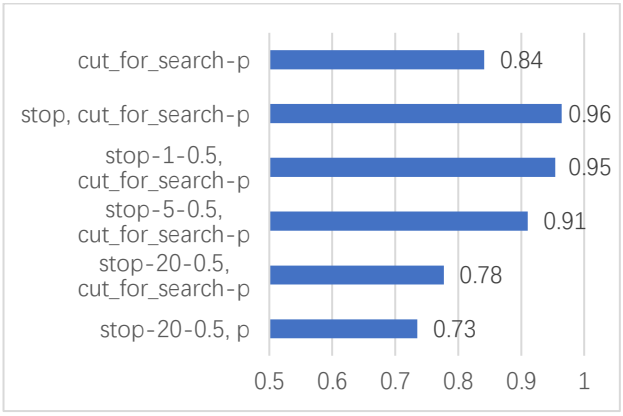

图 6 不同分词策略下的平均准确度

Fig. 6 Average accuracy under different word segmentation strategies

其中“stop”表示过滤停用词，“20-0.5”表示过滤掉总词频小于 20 的词语，及总词频大于 50%的词语；“cut_for_search”表示允许对短文本查询中的同一汉字重复分词，不遗漏分词组合。

实验结果表明，过滤低频词会影响冷门文档的语义搜索效果；过滤高频词和停用词不影响语义搜索效果，但会导致训练同等准确度的 LDA 模型耗费更多时间；查询文本中的错误中文分词会影响语义搜索效果，考虑所有分词路径能获得更好的效果。

3.6 模型效果

最终模型选择过滤停用词，主题数为 300，允许对短文本查询中的同一汉字重复分词，采用生成概率作为相似度度量。最终模型的平均准确度为 0.9934，对于大部分短文本查询可以取得较好的效果。以下为一些查询结果示例：

	score	title		score	title
909	0.082623	Windows XP	719	0.000080	计算机语言
361	0.054110	Windows 98	145	0.000076	编程语言
364	0.046134	Microsoft Windows	50	0.000023	计算机程序
518	0.042010	Windows 2.0	371	0.000022	NTFS
366	0.041702	Windows 2000	67	0.000021	计算语言学
363	0.041036	Windows NT	486	0.000021	第一代编程语言
359	0.040210	Windows 3.1x	330	0.000020	阿基米德
360	0.039409	Windows 95	49	0.000017	程序设计
318	0.034390	Windows 1.0	40	0.000015	操作系统
317	0.033375	微软	31	0.000015	数据结构
841	0.029918	文件扩展名	391	0.000012	Forth
384	0.029840	Microsoft Office	432	0.000012	信息管理系统
362	0.024675	蓝屏死机	4	0.000011	计算机科学
473	0.018100	DirectX	975	0.000010	软件
40	0.014506	操作系统	246	0.000010	数学家
250	0.013086	Delphi	704	0.000009	算法
365	0.011742	DOS	836	0.000009	聂耳
475	0.011730	OpenGL	357	0.000009	多用户
300	0.010910	Visual Basic	273	0.000008	自然语言
746	0.009846	电视	30	0.000007	计算

图 7 “Windows”和“计算机”的搜索结果

Fig. 7 Search results for "Windows" and "Computer"

	score	title		score	title
246	0.036766	数学家	71	0.026731	华侨华人
23	0.018464	心理学	302	0.026729	中国人
57	0.013222	运算数学	539	0.023569	知识产权
136	0.012739	离散数学	986	0.020306	约翰·哈比森
0	0.012165	数学	861	0.018618	洛杉矶
432	0.011743	信息管理系统	105	0.013442	黑龙江省
990	0.010759	生物学家	831	0.012939	中国朝代
285	0.010258	空间科学	370	0.012937	朝鲜的称号
138	0.010184	数理逻辑	96	0.012691	安徽省
14	0.007001	信息科学	107	0.012543	湖北省
825	0.005620	邪教	115	0.012486	四川省
163	0.005544	严重急性呼吸系统综合症	32	0.012248	中华人民共和国
287	0.005520	汉斯·莫拉维克	149	0.012044	法国
665	0.005188	家政学	791	0.011602	武则天
425	0.005168	皮埃尔·西蒙·拉普拉斯	721	0.011364	华国锋
883	0.004730	允禵	446	0.011356	四人帮
741	0.004482	AutoCAD	101	0.011350	中华人民共和国各省级行政区人口列表
592	0.004433	工程学	389	0.011348	江泽民
724	0.004380	氧	60	0.011196	中华人民共和国历史
451	0.004241	弗兰西斯·培根	516	0.011147	胡锦涛

图 8 “数学”和“中国”的搜索结果

Fig. 8 Search results for “Mathematics” and “China”

4 结 论

本文研究了基于潜在狄利克雷分配模型的语义搜索方法的实际效果，使用短文本查询对不同相似度度量、分词策略和主题数下的模型效果进行了对比和分析，提供了一种对中文语料库进行语义搜索的可行方法。

在本文的实验中，最佳主题数是通过训练不同主题数的 LDA 模型后，手动进行选取的。因此一个后续改进方向是实现主题数的自动选取，降低模型的训练成本。另一点则是本文的实验只使用了 1000 个词条，未对不同语料规模下的搜索效果进行测试。

此外，还可将基于 LDA 的语义搜索方法与基于 BERT（Bidirectional Encoder Representations from Transformers）(Devlin et al. 2019) 和 word2vec (Mikolov et al. 2013) 等词嵌入（word embedding）模型的语义搜索方法进行比较。也可将词嵌入模型与 LDA 模型结合，替换 LDA 所使用的整数词编码，例如使用 lda2vec (Moody 2016) 模型。

参考文献(References)

Blei, David M. 2003. “Latent Dirichlet Allocation,” 30.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” arXiv. https://doi.org/10.48550/arXiv.1810.04805.

Goodwin, Phil. 2019. “Tape and Cloud: Solving Storage Problems in the Zettabyte Era of Data,” June.

Griffiths, Thomas L., and Mark Steyvers. 2004. “Finding Scientific Topics.” *Proceedings of the National Academy of Sciences* 101 (suppl_1): 5228–35. https://doi.org/10.1073/pnas.0307752101.

Kuo, Carbo. (2010) 2010. “Open Chinese Convert.” C++. https://github.com/BYVoid/OpenCC.

Lin, J. 1991. “Divergence Measures Based on the Shannon Entropy.” *IEEE Transactions on Information Theory* 37 (1): 145–51. https://doi.org/10.1109/18.61115.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. “Efficient Estimation of Word Representations in Vector Space.” arXiv. https://doi.org/10.48550/arXiv.1301.3781.

Moody, Christopher E. 2016. “Mixing Dirichlet Topic Models and Word Embeddings to Make Lda2vec.” arXiv. http://arxiv.org/abs/1605.02019.

Řehůřek, Radim, and Petr Sojka. (2011) 2010. “Software Framework for Topic Modelling with Large Corpora.” Python. Proceedings of LREC 2010 Workshop New Challenges for NLP Frameworks. Valetta, MT: University of Malta. http://is.muni.cz/publication/884893/en.

Sun, Junyi. 2012. “Jieba: Chinese Word Segmentation Tool.” https://github.com/fxsjy/jieba.

“Zhwiki Dump Progress on 20221201.” 2022. December 1, 2022. https://dumps.wikimedia.org/zhwiki/20221201/.

Semantic search method based on latent Dirichlet allocation

model

ZHANG Tong[1]

1. *School of Computer Science, China University of Geosciences (Wuhan), Wuhan 430078, China*

Abstract: The traditional keyword-based text information retrieval methods lack the ability to search the semantics of text. In this paper, we use Latent Dirichlet allocation model and Gibbs sampling method to model text data, get the topic distribution of text, and use it to realize semantic search of text data. Experiments were conducted using short text queries to compare and analyze the effect of the model under different similarity measures, word segmentation strategies and topic numbers. The results showed that the method could achieve high search accuracy.

Key words: Information retrieval, Topic modeling, Latent Dirichlet allocation(LDA),Gibbs sampling