计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(6):151−157 [doi: 10.15888/j.cnki.csa.006394]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于张量分解的分布式主题分类模型①

马年圣[1], 卞艺杰[1], 唐明伟[2]

[1](河海大学 商学院, 南京 211100)
[2](南京审计大学 管理科学与工程学院, 南京 211815)
通讯作者: 马年圣, E-mail: MacMargo@163.com

摘　要: 针对大规模数据分类时计算时间长以及分类精度下降等问题, 提出使用张量分解求解 LDA 主题模型参数, 实现对海量网络数据的采集、分类、挖掘. 该方法使用矩量法将 LDA 模型求解转化为低维的张量分解问题, 通过分解和反射进行参数的传递, 运用大数据平台 Spark 的进行分布式计算. 实验结果表明, 改进的模型参数计算方法在时间效率和困惑度方面都得到了提升, 并且分类信息更加直观, 更加适用于大规模网络数据分类工作.

关键词: LDA 主题模型; 张量分解; Spark; 数据分类

引用格式: 马年圣,卞艺杰,唐明伟.基于张量分解的分布式主题分类模型.计算机系统应用,2018,27(6):151–157. http://www.c-s-a.org.cn/1003-3254/6394.html

Improved Distributed Topic Classification Model Based on Tensor Decomposition

MA Nian-Sheng[1], BIAN Yi-Jie[1], TANG Ming-Wei[2]

[1](Business School, Hohai University, Nanjing 211100, China)
[2](School of Management Science and Engineering, Nanjing Audit University, Nanjing 211815, China)

Abstract: Aiming at the problems of large computation time and low classification time, this study presents an improved parameter estimation model for LDA by using the method of tensor decomposition, which can collect, classify, and mine massive network data. Using the method of moments, the LDA model calculation is transformed into low-dimensional tensor decomposition, and the parameters are transferred by decomposition and reflection. The large data platform Spark is used for distributed computation. The experimental results show that the model has been improved in terms of running time and perplexity, and the classification information display is more intuitive, which is more suitable for large-scale network data classification.

Key words: LDA theme model; tensor decomposition; Spark; data classification

大数据时代, 网络信息纷繁复杂, 需要我们从众多网络数据中提取出高价值的隐含信息, 挖掘出的分类信息可用于内容推荐、针对性营销以及实时预测等功能. 而其中主题分类又是现今网络信息时代的一大研究热点, 传统的主题分类主要是以基本分类方法以及人工标签来实现, 但是人工干预过多势必影响到最终的分类结果, 这就需要我们寻求一个无监督的方法, 从文档信息的采集到最后的结果输出无需人工参与.

LDA (Latent Distributed Allocation) 主题模型便是一个无监督的数据挖掘方法, 该模型可从大规模数据中进行文档主题的抽取, 能够出色地完成挖掘文本的潜在关系、判别关联性等工作, 显著提高信息的分类及利用效率. LDA 模型参数计算的空间以及时间复杂度较高, 并且对软硬件需求也提出高要求, 所以模型参数求解优化一直是研究热点. Blei 等人采用“变分推断-EM”算法进行 LDA 模型参数计算, 在单机模式下, 随

① 基金项目: 国家自然科学基金青年项目 (71603114); 江苏省社会科学基金青年项目 (16TQC004); 中国博士后基金面上项目 (2015M581776)
收稿时间: 2017-10-09; 修改时间: 2017-11-01; 采用时间: 2017-11-10; csa 在线出版时间: 2018-05-28

机变分推断快速而准确, 但是在分布式计算中因交互过高而显疲态[1]; 批量变分推断具有很高的交互效率, 但在计算 E-step 时并行效率差强人意[2]; 马尔可夫链在分布式同步和异步计算方面体现出较好的移植性, 但其计算效率过低还有待优化[3]; 唐晓波等人采用热度进行模型参数计算的优化, 通过求解微博的热度来实现信息的分类工作, 其结果也更加直观, 但是其热度的计算方法比较单一, 并不适用于其他的网络数据的分类工作[4].

而在 LDA 模型的针对性使用方案方面也进行了大量研究, Ramage 等人提出 Labeled LDA 模型进行有监督的主题分类, 在主题建模中添加文档的标签, 克服了原始模型强制分配主题的缺陷, 但是也使得计算量翻倍增加[5]; 桂思思等人融入多时间节点函数进行用户兴趣的预测, 但是时间差值的确定比较主观, 偏差不可避免[6]; 关鹏等人采用生命周期理论同主题模型结合, 能够展现所观察文本的随时间所发生的变化, 然而参数的计算没有改进为适合生命周期理论的方法[7].

上述国内外对于 LDA 主题模型的改进都针对特定的数据分类, 而在处理数据量大、维度较高的网络信息时效率、准确性等问题便凸显出来, 且上述研究大部分都是单机下进行实验, 平台移植性较差. LDA 主题模型涵盖了大量的数据以及变量, 构成了高维数据问题, 在时间轴上产生了大量的多元数据, 其中也包含很多数据噪声, 而张量分解方法能够通过数据降维以及张量近似的方法来优化计算. 本文通过随机奇异值分解和白化变换将主题模型参数计算转化为三阶张量的 CP 分解, 加之以 ALS 算法以及数据处理技术, 极大地提高了并行化和准确性, 可达到更高的收敛率以及抗干扰性. 本文实验在 Spark 集群上进行, 充分发挥 Spark 作为轻量级大数据处理框架的特点, 及其大规模数据的计算效率明显优于 Hadoop 的特性. 改进后的 LDA 计算模型适用于大数据时代复杂且高维的信息特点, 能够出色地完成巨量网络信息的分类工作, 适用于搜索引擎、文本解读、信息推送等数据应用领域.

1 相关基础理论

在国内外学者的讨论当中, LDA 主题模型暴露出其不足的方面, 单机模式下, 模型训练时间长, 精确度不高, 并且对于模型超参求解的要求较高, 这些都对模型的发展应用提出了挑战. 现被广泛使用的 LDA 参数求解方法有变分推断和马尔可夫链, 但数据量较大的情况下, 两种方法的计算效率还是比较低下, 这就需要我们采用“分治”思想, 选用张量分解的方法来优化模型参数计算, 采用更高效率和精确度的降维计算方法, 同时使用分布式计算模式来提升模型训练的效率, 以适用于网络大数据量文本的主题分类推荐.

1.1 LDA 主题模型

潜在狄利克雷分布模型 LDA 由 Blei 等人于 2003 年提出后, 便被广泛应用于观点挖掘、主题相关性和信息检索等领域[8]. LDA 通过对离散数据集的建模, 从中提取文本隐含主题, 能在海量网络数据中自动寻找信息间的语义主题, 克服传统信息检索中文档相似度计算方法的缺陷. LDA 主题模型属于词袋模型, 它认为文本中包含着无序的词语, 参数空间的规模与训练文档数量无关, 适合处理大规模语料库. 同时作为全概率生成模型, LDA 主题模型的突出优点是具有清晰的层次结构[9], LDA 是一个三层的贝叶斯框架模型, 每一层都有相应的随机变量或者参数控制, 包含词汇、主题、文档的三层结构, 数据集中的文档被看作是有限个隐含主题所构成的混合分布, 而相应的每个主题也都是对应的数据集中一组特征词汇的混合分布, 模型的概率图如图 1 所示.

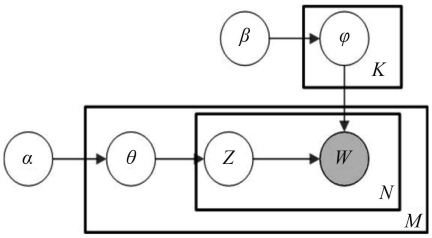

图 1 LDA 主题模型概率图

图 1 中, 只有W是可观察到的变量, 其他都是隐含变量或者参数. 其中, φ表示“主题-词语”分布, θ表示“文档-主题”分布, α、β分别是θ和φ的先验分布, N表示文档的单词总数, M表示文档的总数, Z为选定的主题, 由以上 LDA 主题模型概率图可得到主题生成的联合概率如公式 (1) 所示:

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

LDA 模型训练便是求得参数α和β的值, 使$P(\theta|\alpha,\beta)$为最大. 同 LSA 和 PLSA 模型会产生的过拟合问题不同, LDA 主题模型采用狄利克雷分布, 从而简

化了模型的推导过程, 并且具有很好的先验概率假设, 参数数量不会随着文本数量的增长而线性增长, 泛化能力强, 在算法复杂度和展示效果方面表现优越, 广泛应用于文本的处理当中.

1.2 CP 分解

CP 分解, 即 Candecomp/Parafac 分解, 是传统矩阵分解的拓展, 广泛应用于信号传输、数据分析等领域, 它是把张量分解为一系列 rank-one 张量的计算过程, 对于一个三阶张量$\chi \in \mathbb{R}^{I\times J\times K}$, CP 分解可以写成如下的向量和的形式:

$$\chi = \sum_{r=1}^{R} a_r \otimes b_r \otimes c_r \tag{2}$$

其中, $\otimes$表示张量积运算, R 表示张量的秩, $a_r \in \mathbb{R}^I$, $b_r \in \mathbb{R}^J$, $c_r \in \mathbb{R}^K$, $r = 1,2,\cdots,R$. 公式 (2) 中三阶张量也可写成如下元素乘和的等价形式:

$$\chi_{ijk} \approx \sum_{r=1}^{R} a_{ir} b_{ir} c_{ir} \tag{3}$$

式中, $i = 1,2,\cdots,I$, $j = 1,2,\cdots,J$, $k = 1,2,\cdots,K$. 根据公式 (3), CP 分解便将张量表示为有限数目的 rank-one 张量之和, 分解模型如图 2 所示.

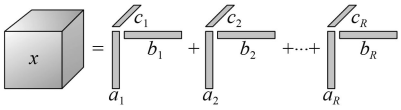

图 2 CP 分解模型

CP 分解具有唯一性, 其实质上指的是张量的秩分解是唯一的, 而传统的矩阵分解并不是唯一的[10]. 目前已有多种方法可以计算 CP 分解, 其中最简单有效的是交替最小二乘法 (Alternating Least Square, ALS), 也是本文所选用的张量分解方法. 对于三阶张量$\chi \in \mathbb{R}^{I\times J\times K}$, ALS 的思想是找到 R 个 rank-one 张量或者一组因子矩阵 (A, B 和 C) 来逼近χ[11]. 如公式 (4) 所示:

$$\begin{aligned} &\text{迭代目标}: \quad \min_{A,B,C} \left\| \chi - (C \odot B) A^{\mathrm{T}} \right\|; \\ &\text{初始化}: \quad B = \hat{B} \text{和} C = \hat{C}; \\ &\text{迭代}: \quad \hat{A}^{\mathrm{T}} = \left(\hat{C} \odot \hat{B}\right) X^{(1)} \\ &\qquad\quad \hat{B}^{\mathrm{T}} = \left(\hat{A} \odot \hat{C}\right) X^{(2)} \\ &\qquad\quad \hat{C}^{\mathrm{T}} = \left(\hat{A} \odot \hat{B}\right) X^{(3)} \end{aligned} \tag{4}$$

式中, 符号⊙表示 Khatri-Rao 积, 当满足一定的迭代条件时, 迭代终止. 因为 ALS 算法需多次迭代才收敛, 所以我们将算法应用到 Spark 平台中进行分布式计算, 以求快速的求得全局的最优参数, 减少大量的实验时间, 这也是分布式计算在现今模型求解中的优势之处.

2 基于张量分解的主题分类模型

2.1 基于张量分解的 LDA 主题分类主体模型

在 LDA 主题模型中, 每篇文档都存在着 K 个潜在的主题, 第 k 个主题具有“主题-词语”的条件分布概率φ_k, 将所有主题的条件分布概率组成矩阵$\varphi = [\varphi_1, \varphi_2, \ldots, \varphi_K] \in \mathbb{R}^{V\times K}$, V为总词汇量, 则φ便是模型求解的“主题-词语”分布矩阵. 而在第m篇文档中, 其混合分布的潜在话题是根据狄利克雷先验参数$\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_k] \in \mathbb{R}^K$所求得, 已知先验参数分布下, 便可求得文档$m$的“文档-主题”分布矩阵$\theta_m$.

传统的 LDA 主题模型的参数估计方法包括变分推断, 马尔可夫链等, 本文采用矩量法将参数估计转化为张量分解的方式进行迭代. 主题为K的 LDA 主题模型可通过文本词汇表示为张量的形式, Anandkumar 等人[12]对主题模型张量的表现形式有如下定义.

$$M_1 := E[x_1] \tag{5}$$

$$M_2 := E[x_1 \otimes x_2] - \frac{\alpha_0}{\alpha_0 + 1} M_1 \otimes M_1 \tag{6}$$

$$\begin{aligned} M_3 :=& E[x_1 \otimes x_2 \otimes x_3] - \frac{\alpha_0}{\alpha_0 + 2}\Big(E[x_1 \otimes x_2 \otimes M_1] \\ &+ E[x_1 \otimes M_1 \otimes x_3] + E[M_1 \otimes x_2 \otimes x_3] \\ &+ \frac{2\alpha_0^2}{(\alpha_0 + 2)(\alpha_0 + 1)} M_1 \otimes M_1 \otimes M_1\Big) \end{aligned} \tag{7}$$

其中, $x \in \mathbb{R}^V$表示一个词语, V为文档集中所有的词汇, x_1, x_2和x_3为同一篇文章的词语, 对于词语v, 任意$u \neq v$, $x_v = 1$, $x_u = 0$. 符号⊗为张量积运算, 任意的$x \otimes x = xx^{\mathrm{T}}$, E为向量期望, $\alpha_0 = \{\alpha_k\}_{k=1}^K$, 表示话题分布的稀疏程度, α_0越小, 表明文档中隐含的主题越少. 张量M_2、M_3通过分解可转化为如下张量积的表现形式:

$$M_2 = \sum_{k=1}^{K} \alpha_k \varphi_k \varphi_k \tag{8}$$

$$M_3 = \sum_{k=1}^{K} \alpha_k \varphi_k \varphi_k \varphi_k \tag{9}$$

其中, K为我们从文档集中抽取的主题数, 通过公式

(5)~(9), 主题模型参数求解便可转化为矩阵张量分解的方式. 从公式 (8) 可以得出二阶矩M_2的低秩分解可求得包含α_k和φ_k的子空间, 而M_3的张量分解可求得潜在狄利克雷先验分布α以及“主题-词语”分布矩阵φ, 最终通过先验分布α求解“文档-主题”矩阵θ.

在进行M_3张量分解分解前, 通过数据的预处理 (包括数据向量化、正交化和降维操作等) 来保证模型的收敛率和抗噪声干扰, 随机奇异值分解[13]作为高效的矩阵低秩分解手段, 此处选用该方法来执行对M_2的正交分解, 接下来利用矩量法将 LDA 主题模型参数估计转化为低维下张量的 CP 分解, 最终生成“文档-主题”、“主题-词语”矩阵. 模型参数求解步骤如表 (1) 所示.

模型最终会生成“文档-主题”、“主题-词语”概率分布, 根据“文档-主题”矩阵可选取概率最大的主题为该文档的第一候选主题, 而通过“主题-词语”矩阵可推断是该主题的具体含义, 结合文档中已经得出的候选主题, 便可实现该文档的主题分类.

表 1 基于张量分解的主题分类模型求解步骤

Begin.

1. $D\in\mathbb{R}^{M\times N}\leftarrow preprocess(doc)$ //数据采集、分词、预处理
2. $StartLDA\left(K,\alpha_0,D\in\mathbb{R}^{M\times N}\right)$
3. $ComputeM_2=M_2(D)$ //$M_2\in\mathbb{R}^{V\times V}$
4. $U,\Sigma,V\leftarrow Randomized\ SVD(M_2,K)$ //$U\in\mathbb{R}^{V\times K}$
5. $M_3\leftarrow M_3(U,\Sigma)$ //$M_3\in\mathbb{R}^{K\times K^2}$
6. $(\lambda,A)\leftarrow CP\_ALS(M_3,K)$
7. $\{\varphi_k\}_{k=1}^{K}\leftarrow resotre(A)$ //生成“主题-词语概率矩阵”
8. $\alpha_k\leftarrow\lambda_k^{-2}$ //生成$\{\alpha_k\}_{k=1}^{K}$
9. $\{\theta_m\}_{m=1}^{M}\leftarrow \mathrm{LDA}\left(\{\alpha_i\}_{i=1}^{k},D\in\mathbb{R}^{M\times N}\right)$ //“文档-主题”概率矩阵
10. $Sort\{\theta_m\}_{m=1}^{M}\{\varphi_k\}_{k=1}^{K}$ //矩阵概率排序, 选取词语前 20 列
11. $Classify\{\theta_m\}_{m=1}^{M}$ //文档主题分类

End.

2.2 模型的关键技术

2.1 小节中基于张量分解的 LDA 主题分类模型可拆分为 3 个重要阶段, 第 1 阶段为数据预处理, 第 2 阶段为基于 ALS 算法的 CP 分解, 第 3 阶段为主题分类计算.

(1) 数据预处理

网络信息不同于普通文本信息, 数据形式、结构均有差异, 所以预处理的首要工作便是进行分词等一系列操作, 数据预处理完成后, 需对数据进行向量化以及降维操作, 以便大量减少参数迭代时的计算量. 在进行张量形式的多维数组操作时, 数据维数的大小直接决定了矩阵操作的计算量大小, 尤其是在处理自然语言这种高维数据时, 在内存中进行三阶矩的存储操作的运算量都是极大的. 数据稀疏化是其中一类方法, 更好的则是进行线性降维, 加之以张量乘积的形式来避免直接生成张量, 能够大幅度减少计算规模, 并且对于张量的操作也是高效的[14].

在此首先进行张量白化变换 (Whitening Transformation), 低秩正交分解二阶矩M_2. 奇异值分解在进行矩阵分解中表现出极大的优势, 但当数据的行列数过大时, 奇异值分解表现出分解缓慢、效率低等缺点, 而随机奇异值分解通过生成子空间进行迭代运算能够加快分解工作, 此处采用随机奇异值分解进行M_2的分解操作[13].

随机奇异值分解算法可以总结为两步计算, 第一阶段构造一个正交基, 其值域接近于M_2, 即构造正交矩阵Q, 使得$M_2\approx QQ^{\mathrm{T}}M_2$; 第二阶段将矩阵$M_2$约束于$K$维子空间中, 运用奇异值分解来计算$Q^{\mathrm{T}}M_2$, 求得$U$、$\Sigma$、$Z$.

由随机奇异值分解可得$M_2=U\Sigma U^{\mathrm{T}}$, 定义$W:=U\Sigma^{-0.5}\in\mathbb{R}^{d\times k}$为白化矩阵, 令$\hat{\varphi}=\mathrm{Diag}\left(\alpha_i^{0.5}\right)W^{\mathrm{T}}\varphi$, 则$\hat{\varphi}\in\mathbb{R}^k$便是正交向量, 证明如下:

$$\begin{aligned}\hat{\varphi}\hat{\varphi}^{\mathrm{T}}&=\Sigma^{-0.5}U^{\mathrm{T}}\mu\mathrm{Diag}\left(\alpha_i^{0.5}\right)\mathrm{Diag}\left(\alpha_i^{0.5}\right)\mu^{\mathrm{T}}U\Sigma^{-0.5}\\&=\Sigma^{-0.5}U^{\mathrm{T}}U\Sigma U^{\mathrm{T}}U\Sigma^{-0.5}=W^{\mathrm{T}}M_2W=I_K\end{aligned}\tag{10}$$

最后使用公式 (7) 可计算生成维数为K^3的正交三阶矩$\tilde{M}_3=M_3(W,W,W)=\sum_{k=1}^{K}\alpha_i^{-0.5}\hat{\varphi}_k\hat{\varphi}_k\hat{\varphi}_k$, 至此, 便完成了$M_3$白化以及正交化操作, 即数据预处理阶段结束.

(2) 基于 ALS 算法的张量分解

$\tilde{M}_3$计算生成后, 运行基于交替最小二乘法的张量分解, ALS 算法的核心是找到最接近M_3的有限数目的 *rank-one* 之和[11], 即为:

$$\min_{\hat{M}_3}\tilde{M}_3-\hat{M}_3\leftarrow\hat{M}_3=\sum_{i=1}^{K}\lambda_i a_i b_i c_i=\lambda;A,B,C\tag{11}$$

其中, $\hat{M}_3$为分解的 *rank-one* 之和, 交替最小二乘法是一个迭代算法, 算法交替的进行A, B, C的优化, 每一次迭代过程中, 总是假定其他两个矩阵是已知的, 通过求解最小化的问题来分解矩阵. 当B和C值固定后, 可以将

公式改写为如下形式:

$$\min_{\hat{X}} \tilde{M}_3 - \hat{X}\left((C\odot B)^{\mathrm{T}}\right) \text{ s.t. } \hat{X} = X\cdot Diag(\lambda) = \tilde{M}_3\left[(C\odot B)^{\mathrm{T}}\right]^{\dagger} \tag{12}$$

将$\hat{X}$带入最小值求解中, 最终基于交替最小二乘法的张量分解便转化为如下的最优化计算:

$$(\lambda, A) \leftarrow \min_{\lambda\in\mathbb{R}^k, X\in\mathbb{R}^{k\times k}} X\cdot Diag(\lambda)(C\odot B)^{\mathrm{T}} - \tilde{M}_3 \quad \text{s.t. } \forall k: X_k = 1, \lambda_k \geq 0 \tag{13}$$

其中, $\odot$表示 Khatri-Rao 积, 每次迭代都进行λ的计算以保证特征向量每一列均为归一化, 此处采用 Khatri-Rao 积的伪逆矩阵形式优化计算[15], 如公式 (13) 所示:

$$(C\odot B)^{\dagger} = \left(\left(C^{\mathrm{T}}C\right) * \left(B^{\mathrm{T}}B\right)\right)^{\dagger}(C\odot B)^{\mathrm{T}} \tag{14}$$

式中, $*$为哈达马乘积, 通过变换, 仅需计算$K\times K$的伪逆矩阵而无需计算$K\times K^2$原矩阵. ALS 算法是一种批量同步并行计算模型[16], 在 K 阶并行的保证下, 公式 (11) 中左边的每一行均可作为M_3独立的一部分来进行参数的估计, 并且在使用 *Spark* 计算框架进行分析时, 每运行一个 ALS 子程序之前可通过广播变量同步最新估计的参数[17], 进行算法迭代时的空间需求以及每个节点所进行的总交互量均为$O\left(K^2\right)$.

(3) 模型主题分类计算

张量分解收敛后, 采用反白化变换, 计算原文档集中的狄利克雷先验分布α以及“主题-词语”分布矩阵φ. 反白化变换强调张量结构的特殊性[12], 通过分解后的张量数据来投影反射出 LDA 模型参数, 如下所示:

给定 CP 分解后的$\tilde{M}_3$, 向量$\mu_1, \mu_2, \cdots, \mu_k \in \mathbb{R}^V$线性无关, 标量$O_1, O_2, \cdots, O_K$均大于 0, 则:

①$\tilde{M}_3$的特征值和特征向量分别为$\{O_k\}_{k=1}^K$和$\{\mu_k\}_{k=1}^K$.

②原词汇空间的狄利克雷先验参数$\alpha = \left\{O_k^{-2}\right\}_{k=1}^K$.

③$\left(W^{\mathrm{T}}\right)^{\dagger}$是$W^{\mathrm{T}}$的穆尔彭罗斯伪逆矩阵[18], 原词汇空间的“主题-词语”分布概率$\varphi = \left\{O_k\left(W^{\mathrm{T}}\right)^{\dagger}\mu_k\right\}_{k=1}^K$.

由反白化变化可推导出$\alpha = \left\{\frac{1}{\sqrt{O_k}}\right\}_{k=1}^K$, 同时给定分解后的特征向量$\mu$, 求解矩阵$\varphi\in\mathbb{R}^{V\times K}$使得$\varphi \approx \left\{w_k\left(W^{\mathrm{T}}\right)^{\dagger}\mu_k\right\}_{k=1}^K$. 待原词汇空间参数求解后, 根据原输入文档集和先验分布$\{\alpha_i\}_{i=1}^k$, 生成“文档-主题”分布矩阵$\{\theta_m\}_{m=1}^M$, 最后, 为了更直观的显示以及更精准的分类, 将“文档-主题”、“主题-词语”矩阵进行概率排序, 在进行文档分类时需指定特定的分类类别, 所以我们根据文档中的重点主题以及主题中的重点词语, 选取其中概率最高主题为该文档的主题类别, 抽取概率为前 20 的词语作为该主题的特征词, 进行下一步的主题分类工作.

3 仿真实验

3.1 平台构建

实验包括模型对比和主题分布分析, 实验数据通过 *WebMagic* 爬虫技术在网络上自动抓取, 通过对页面的分析来下载相应的新闻信息文本, 主要采集于各大新闻网站的新闻信息数据, 如“中国新闻网”、“凤凰网”等, 主要涉及经济、军事、文化等领域, 在进行文本的白噪声处理后, 筛选出 1800 条作为原始分析数据. 为保证实验的可靠性以及可识别性, 需定义停用词表, 词表中包含常用词、常见语气词、助词等高频率出现的词语, 同时根据中文文本的特殊性, 还进行了繁简转换, 保证实验数据的格式统一, 通过该停用词典可剔除大部分的噪声词语[19].

实验使用 Scala 作为编程语言, 在 Spark 集群模式上进行模型训练与预测, 主节点 *master* 进行任务调度, 从节点 *worker* 进行同步的运算. *worker* 之间交替的计算更新的参数, 广播参数至其他的节点, 最后进行数据的同步. 而 *master* 则负责检查是否实时的检验是否需要结束运算以及负责各节点资源之间的调度, 实验集群均为 Centos 7 系统, 每个节点内存均为 4 G, 实验主要步骤如图 3 所示.

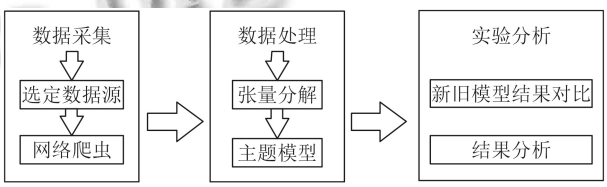

图 3 仿真实验步骤

3.2 实验结果与分析

实验首先将模型训练时间和困惑度同基于 EM 算法的 LDA 模型进行对比, 其中, 模型生成时间是体现模型计算是否高效的重要指标之一, 而困惑度则是衡量模型是否同原始数据相吻合的重要检验标准, 最后通过网络新闻数据的预测, 来说明基于张量分解的 LDA 主题模型适用于网络数据的分类工作.

(1) 训练时间对比

在相同运行环境下, 设置迭代次数为 500 次, 主题

数为 50, 将本文模型同基于 EM 算法的主题模型进行训练对比, 通过增加计算节点数来对比模型训练时间长短, 结果显示基于张量分解的主题模型在时间方面显现出极大的优势, 如图 4 所示.

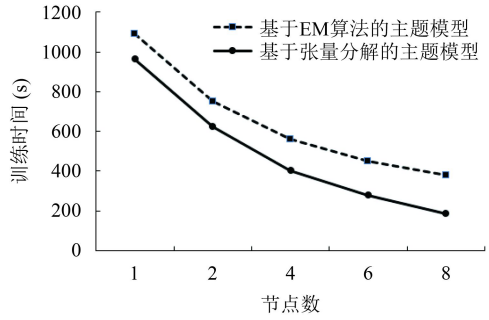

图 4 模型时间对比图

从图中可以看出, 基于张量分解的主题模型在训练时间明显优于基于 EM 算法的 LDA 主题模型. 增加节点数对于运算时间的减少是明显的, 体现出 Spark 大数据平台在各节点内存不变的情况下, 节点个数对于运行时间是成反比的. 两个算法开始增加节点数对于时间的优化更是相当显著, 但随着节点数的增加, 增益效果降低, 同基于 EM 算法的 LDA 主题模型相比, 基于张量分解的 LDA 模型在节点数增加时, 其计算时间下降幅度更大, 表明基于张量分解的 LDA 主题模型对多节点的集群有更好的计算能力, 更加表现出模型对于大运算量的适应性.

(2) 困惑度对比

困惑度作为文本建模中常用的评价指标, 其值越小, 模型对于上下文的约束能力就越强, 表明语言模型吻合度越好[8]. 其公式如下所示:

$$perplexity(D_{\text{test}}) = \exp\left\{-\frac{\sum_{m=1}^{M}\log p(W_m)}{\sum_{m=1}^{M}N_m}\right\} \quad (15)$$

式中, D_{test}为测试文档集, W_m为测试 m 文档中观测到的单词, $P(W_m)$为模型产生文本W_m的概率, N_m为文档 m的单词数.

在相同的语料和参数设置下, 计算基于 EM 算法的 LDA 主题模型和基于张量分解的主题模型, 两种方法困惑度随隐含主题数目的变化情况如图 5 所示.

通过图 5 可得到, 随着主题数量的不断增加, 两个模型的困惑度都在相应的降低, 在达到最低点时, 主题抽取的个数各不相同, 基于张量分解的 LDA 主题模型在该训练文档集中主题数为 50 时困惑度最小. 在数据量较大、主题较多时, 本文模型困惑度明显低于基于 EM 算法的 LDA 主题模型.

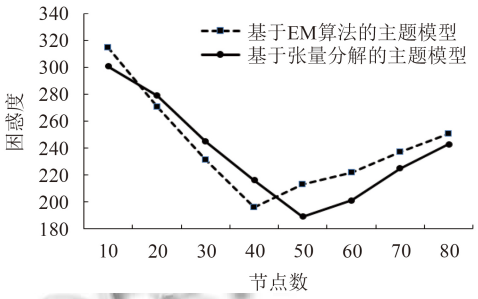

图 5 抽取主题数的困惑度对比

(3) 主题分布分析

将预处理的新闻信息通过本文 LDA 主题分类模型进行训练, 针对新闻文本的特殊性, 在定义特征词时, 进行数据预处理时加入了时间等词的停用, 设置主题数为 50, $\alpha_0 = 1$, 待模型预测完成后, 随机抽取三个文档以及他们相对应的主题进行分析, 部分结果如表 2、表 3 所示.

表 2 $topicN = 50$ 时文档与主题的分布概率

文档 5		文档 678		文档 1555	
主题	概率	主题	概率	主题	概率
1	0.777 850	30	0.597 040	48	0.620 750
35	0.013 814	33	0.163 515	38	0.018 569
6	0.010 399	49	0.022 181	40	0.009 092
…		…		…	

表 3 $topicN = 50$ 时主题与词的分布概率

主题 1(企业)		主题 30(经济)		主题 48(电影)	
特征词	概率	特征词	概率	特征词	概率
企业	0.026 150	指数	0.024 327	电影	0.023 970
发展	0.018 561	价格	0.015 616	观众	0.014 071
创新	0.013 304	经济	0.015 146	导演	0.011 466
服务	0.013 171	制造业	0.011 669	演出	0.009 952
产业	0.011 666	百分点	0.011 417	影片	0.008 847
投资	0.011 595	下降	0.010 992	故事	0.008 549
平台	0.010 996	增长	0.009 797	粉丝	0.007 942
…		…		…	

表 2 可以看出, 每篇文档根据文中词语的分布, 不局限于单个主题, 但第一个主题的概率较大, 可以整体概括整篇文档的大概主题方向. 例如文档 5 中主题 1 的概率为 0.777 85, 相对应, 主题一中出现的都是企业发展类的词汇, 则主题 1 便为企业主题, 进一步的将文档 5 便可分类到企业模块.

表 3 清晰地展现出不同主题其中的含义, 可读性

强, 同时本文实证数据来源于网络新闻信息, 从中可窥探社会热点. 主题 1 涉及企业发展, 其大部分的词语均是企业在现代社会发展所重视的方面, 同时也是企业发展中强调的高频词. 而主题 30 则是经济类, 通过各经济词语的罗列, 能够对部分的金融的专业用词有一定的了解, 可运用于新闻定位推送, 同时在新闻里出现, 更能说明媒体以及公众对于经济的关注. 最后主题 48 则为文化产业电影类, 新闻中能够涉及到如下的词语, 说明人们在现今生活高压力下对于电影、文化的关注. 以上的“主题-词语”分布能够说明主题模型对于网络数据分类的高效性, 显性地挖掘网络信息中所蕴含的内涵, 可充分适用于信息推荐、搜索引擎当中.

4 结论与展望

本文将张量分解引入到 LDA 主题模型的训练中, 利用矩量法将数据转换为张量分解的计算形式, 运行基于交替最小二乘法的 CP 分解进行参数迭代, 最后使用网络数据在大数据平台 Spark 中验证分析, 实验表明, 基于张量分解的 LDA 主题模型在网络数据主题、词汇生成方面同基础主题模型更有优势, 更加适用于网络数据主题的分类. 当然, 网络数据的预处理准确性有待提高, 对于主题模型的原始输入以及计算优化是我们下一阶段需要研究的内容.

参考文献

1 Hoffman MD, Blei DM, Wang C, *et al*. Stochastic variational inference. Journal of Machine Learning Research, 2013, 14(5): 1303–1347.

2 Nallapati R, Cohen W, Lafferty J. Parallelized variational em for latent dirichlet allocation: An experimental evaluation of speed and scalability. Proceedings of 2007 Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007). Omaha, NE, USA. 2007. 349–354.

3 Griffiths TL, Steyvers M. Finding scientific topics. Proceedings of the National Academy of Sciences of the United States of America, 2004, 101(S1): 5228–5235.

4 唐晓波, 向坤. 基于 LDA 模型和微博热度的热点挖掘. 图书情报工作, 2014, 58(5): 58–63. [doi: 10.11925/infotech.1003-3513.2014.05.08]

5 Ramage D, Hall D, Nallapati R, *et al*. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. Proceedings of 2009 Conference on Empirical Methods in Natural Language Processing. Singapore. 2009. 248–256.

6 桂思思, 陆伟, 黄诗豪, 等. 融合主题模型及多时间节点函数的用户兴趣预测研究. 现代图书情报技术, 2015, (9): 9–16. [doi: 10.11925/infotech.1003-3513.2015.09.02]

7 关鹏, 王曰芬. 基于 LDA 主题模型和生命周期理论的科学文献主题挖掘. 情报学报, 2015, 34(3): 286–299.

8 Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. Journal of Machine Learning Research, 2003, 3(4/5): 993–1022.

9 李湘东, 胡逸泉, 黄莉. 采用 LDA 主题模型的多种类型文献混合自动分类研究. 图书馆论坛, 2015, 35(1): 74–80.

10 Sidiropoulos ND, Bro R. On the uniqueness of multilinear decomposition of N-way arrays. Journal of Chemometrics, 2000, 14: 229–239. [doi: 10.1002/(ISSN)1099-128X]

11 Kolda TG, Bader BW. Tensor decompositions and applications. SIAM Review, 2009, 51(3): 455–500. [doi: 10.1137/07070111X]

12 Anandkumar A, Foster DP, Hsu D, *et al*. A spectral algorithm for latent dirichlet allocation. Algorithmica, 2015, 72(1): 193–214. [doi: 10.1007/s00453-014-9909-1]

13 Halko N, Martinsson PG, Tropp JA. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review, 2010, 53 (2): 217–288.

14 Anandkumar A, Ge R, Hsu D, *et al*. Tensor decompositions for learning latent variable models. The Journal of Machine Learning Research, 2014, 15(1): 2773–2832.

15 Liu SZ, Trenkler G. Hadamard, khatri-rao, kronecker and other matrix products. International Journal of Information and Systems Sciences, 2008, 4(1): 160–177.

16 Valiant LG. A bridging model for parallel computation. Communications of the ACM, 1990, 33(8): 103–111. [doi: 10.1145/79173.79181]

17 Wang YN, Tung HY, Smola A J, *et al*. Fast and guaranteed tensor decomposition via sketching. Proceedings of 2015 Advances in Neural Information Processing Systems (NIPS). Montreal, Canada. 2015. 991–999.

18 Macausland R. The moore-penrose inverse and least squares[Thesis]. Tacoma, Washington, USA: University of Puget Sound, 2014.

19 冯永, 李华, 钟将, 等. 基于自适应中文分词和近似 SVM 的文本分类算法. 计算机科学, 2010, 37(1): 251–254, 293.