计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(9):280−286 [doi: 10.15888/j.cnki.csa.008677]

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

面向热点话题检测的增量文本聚类算法①

郭　莹[1], 薛　涛[1], 胡伟华[2]

[1](西安工程大学 计算机科学学院, 西安 710600)
[2](西安工程大学 人文社会科学学院, 西安 710600)
通信作者: 薛　涛, E-mail: 1129925519@qq.com

摘　要: 针对传统的 Single-Pass 聚类算法对数据输入顺序过于敏感和准确率较低的问题, 提出一种以子话题为粒度, 考虑新闻文本动态性、时效性和上下文语义特征的增量文本聚类算法 (SP-HTD). 首先通过解析 LDA2Vec 主题模型, 联合训练文档向量和词向量, 获得上下文向量, 充分挖掘文本的语义特征及重要性关系. 然后在 Single-Pass 算法基础上, 根据提取到的热点主题特征词, 划分子话题, 并设置时间阈值, 来确认类簇中心的时效性, 将挖掘的语义特征和任务相结合, 动态更新类簇中心. 最后以时间特性为辅, 更新话题质心向量, 提高文本相似度计算的准确性. 结果表明, 所提方法的 F 值最高可达 89.3%, 且在保证聚类精度的前提下, 在漏检率和误检率上较传统算法有明显改善, 能够有效提高话题检测的准确性.

关键词: Single-Pass; 文本表示; 文本聚类; 文本相似度; 热点话题检测

引用格式: 郭莹,薛涛,胡伟华.面向热点话题检测的增量文本聚类算法.计算机系统应用,2022,31(9):280–286. http://www.c-s-a.org.cn/1003-3254/8677.html

Incremental Text Clustering Algorithm for Hot Topic Detection

GUO Ying[1], XUE Tao[1], HU Wei-Hua[2]

[1](School of Computer Science, Xi'an Polytechnic University, Xi'an 710600, China)
[2](School of Humanities and Social Sciences, Xi'an Polytechnic University, Xi'an 710600, China)

Abstract: As the traditional Single-Pass clustering algorithm is highly sensitive to the input sequence of data and has low accuracy, an incremental text clustering algorithm (SP-HTD) is proposed, which takes subtopics as granularity and considers the dynamics, timeliness, and contextual semantic features of news texts. Firstly, by parsing the LDA2Vec topic model, this study jointly trains the document vectors and the word vectors to obtain the context vectors and thus fully mines the semantic features and importance relationship of the text. Then, on the basis of the Single-Pass algorithm, subtopics are classified according to the extracted hot topic feature words, and the time threshold is set to confirm the timeliness of the cluster center. The mined semantic features and tasks are combined to dynamically update the cluster center. Finally, with the assistance of the time characteristics, the centroid vectors of the topics are updated to improve the accuracy of text similarity calculation. The results reveal that the F value of the proposed method can reach up to 89.3%, and on the premise of ensuring the clustering accuracy, the proposed method has a significantly lower undetected rate and false detection rate compared with those of the traditional algorithm, and thus it can effectively improve the accuracy of topic detection.

Key words: Single-Pass; text representation; text clustering; text similarity; hot topic detection

① 基金项目: 国家社会科学基金 (18XYY010)
收稿时间: 2021-12-07; 修改时间: 2022-01-04; 采用时间: 2022-01-24; csa 在线出版时间: 2022-07-07

1 引言

随着大数据时代的飞速发展, 如何能够快速、及时地从大量的网络新闻信息中发现热点话题已经成为当前研究的热点. 话题检测[1] 作为一种有效的能够自动从大量网络数据流中挖掘重要信息的研究方法, 在信息检索[2]、舆情监督[3]、舆情预测[4] 等方面有着广泛的应用场景. 如监测和把握中国在国际上的受关注领域和程度, 为中国政府调整外交策略和媒体建构海外中国形象献计献策, 具有重要的研究意义.

文本的话题检测任务主要分为文本表示和文本聚类两个重要部分. 在文本表示方面, 文中以 LDA2Vec 主题模型[5] 为基础, 结合 LDA 模型[6] 注重全局文本语义特征和 Word2Vec 模型[7] 注重局部文本语义特征的优势, 将主题向量和词向量融合到同一语义空间中形成嵌入式向量模型, 进而学习主题, 产生的主题词可解释性更强, 更注重上下文语义相似度, 同时也解决了文本特征维度过高的问题. 但文本表示模型仅考虑了提取隐含语义主题的准确性, 没有考虑到全部文本信息, 且话题的凝聚度不高, 由此, 本文在文本表示的基础上, 利用文本聚类算法, 对数据进行热点话题聚类.

采用增量文本聚类思想, 不需要重新对全部数据进行训练, 可以更全面、更高效地对动态实时增长的数据流进行热点话题聚类. 目前广泛应用的增量文本聚类算法如 Single-Pass 算法[8], 由于其实现简单、高效且不需要提前设定聚类类别数量的优势, 被许多学者研究并改进, 文献 [9] 通过对已经标注的话题类别和时间间隔较远的文档类别增加时间参数动态阈值, 证明了不同文档顺序对聚类效果的影响. 文献 [10] 提出了一种通过调整关键词权重降低文本噪声, 将上下文和相似度矩阵相结合的关联模型, 从而提升算法的话题挖掘速度. 文献 [11] 在文本特征词选取时, 以权重系数表达特征词位置, 并引入了子话题判断, 得到了不同粒度的话题聚类效果. 文献 [12] 在余弦相似度的基础上, 考虑从取值和方向两方面改进余弦相似度, 从而提高话题发现的算法正确率. 上述方法在一定程度上提高了话题聚类的精度, 但随着数据规模的增长, 时间复杂度也急剧增长, 尤其针对动态增长的数据流, 话题检测的准确率依然较低, 同时还会影响到相似度计算结果准确率.

鉴于此, 本文提出一种面向热点话题检测的增强文本聚类算法 (Single Pass-hot topic detection, SP-HTD). 以 Single-Pass 算法思想为基础, 从文本表示、文本聚类和相似度计算 3 个方面进行了改进, 并通过爬取并预处理《纽约时报》《泰晤士报》《朝日新闻》等 10 个国际主流媒体中的涉华报道作为数据集, 与多个聚类算法进行对比实验. 结果表明, 在保证聚类精度的前提下, 所提算法能够取得更优的话题检测效果, 可以有效提升聚类算法对新文本的反应能力.

2 SP-HTD 增量文本聚类算法

2.1 问题的提出

传统的 Single-Pass 算法是一种流式处理文本数据的聚类算法, 根据文档输入的先后顺序, 依次比较要输入的新文本数据与已有类簇的文本相似度来进行划分聚类, 不需要每次对整个文档集合重新聚类, 具有实现便捷、易于理解和应用广泛的特点. 它的基本流程是首先将输入的第一篇文档作为话题聚类的首个类簇, 并设定一个初始的文本相似度阈值, 然后计算要加入的新文本数据与已有的各个类簇文档的相似度, 如果该相似度大于初始的相似度阈值, 就把该文本归为当前聚类类簇, 否则以该文档为聚类中心增加一个新类簇, 直到所有的文档数据处理完毕, 结束话题聚类过程. 其处理流程如图 1 所示.

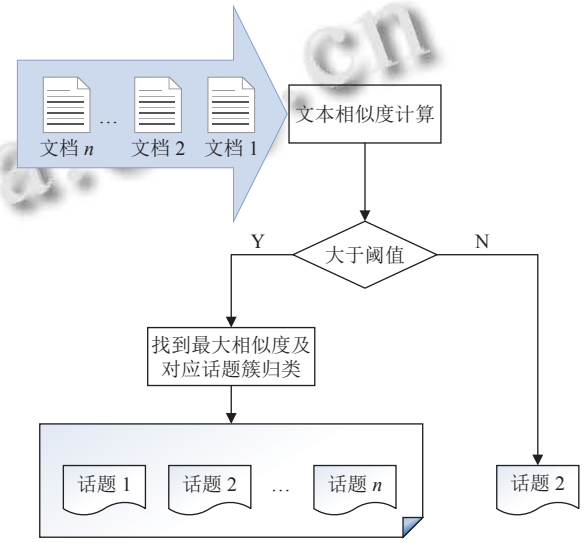

图 1 Single-Pass 算法处理流程

在文本聚类的过程中, Single-Pass 聚类算法对整个文档集合只需要遍历一次, 根据数据实时情况聚类, 不需要给定初始聚类类别的个数, 所以逻辑简单且执行效率高. 但该算法也存在一定的缺陷, 主要体现在以

下两点: (1) 对文本数据的输入顺序过于敏感, 文档的输入顺序会影响文本聚类的结果. (2) 对新文档类簇划分时, 需要逐一比较文本相似度, 随着文档和类簇的增加, 未及时淘汰旧的类簇, 会导致算法计算复杂度增加, 影响聚类效率.

2.2 算法框架

热点话题检测是以话题为粒度, 考虑语料的实时性和数据来源等因素, 利用文本聚类算法去发现新的热点事件, 将同一话题下的新闻报道聚合到同一类簇下, 生成不同的聚类类别, 从而可以更好的组织新闻事件, 了解事件的进展. 处理流程如图 2 所示.

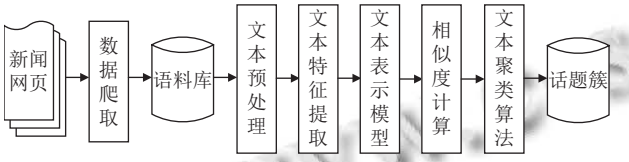

图 2 话题检测处理流程图

本文在文本表示模型的基础上, 改进 Single-Pass 增量文本聚类算法发现新热点话题. 首先通过解析 LDA2Vec 主题模型, 联合训练文档向量和词向量, 获得语料数据的主题分布, 用来解决在文本聚类过程中产生的文本特征维数高和数据稀疏的问题, 然后基于 Single-Pass 算法进行初始化聚类, 引入时间阈值, 确定类簇的时效性, 最后将挖掘的文本语义特征和热点话题检测任务相结合, 动态优化类簇中心, 进行迭代聚类, 并在文本相似度方面, 以新闻报道时间特性为辅, 优化文本相似度计算方法, 改善 Single-Pass 算法的缺陷. 主要改进内容分为文本表示、文本相似度和文本聚类 3 个部分.

2.3 联合训练文本表示

在热点话题聚类过程中, 需要用文本表示模型来表示新闻事件. 传统的 LDA 及其改进模型[13–15] 存在主题语义一致性较弱和准确率较低等问题. 本文依据文献 [16] 提出的 NS-LDA2Vec 主题模型, 在考虑词语信息和主题信息的基础上, 使用 LDA 和 Word2Vec 模型对语料库进行预训练, 然后解析 LDA2Vec 模型的核心算法, 迭代学习语料中含有主题信息的文档向量, 最后联合训练该文档向量与 Word2Vec 训练的词向量得到上下文向量, 利用上下文向量完成热点主题识别任务. 主要分为词向量表示和文档向量表示两个部分.

在词向量表示部分, 根据 Skip-gram 负采样思想[17] 训练得到文本的词向量表示, 采用文献 [7] 提出的移动窗口形式来扫描数据集, 通过对模型多次迭代训练, 对窗口参数进行调优, 文中将滑动窗口的大小设置为 5, 即包含中枢词在内的 5 个单词, 然后动态移动窗口, 利用选定的中枢词来预测邻近窗口内出现的目标词, 从而学习文本的上下文和主题信息, 学习的上下文向量表示表现的更为密集. 文档向量表示部分主要包括文档权重向量和主题向量的计算. 文档权重向量表示文档中各个主题的重要性. 主题向量是通过调节文档权重来更新主题强度. 初始化语料库中文档的权重向量时, 通过约束文档向量$\vec{d}_j$生成一组潜在主题向量$\vec{t}_0,\vec{t}_1,\cdots,\vec{t}_k,\cdots,\vec{t}_n$, 计算公式如式 (1) 所示:

$$\vec{d}_j = p_{j0}\cdot\vec{t}_0 + p_{j1}\cdot\vec{t}_1 + \cdots + p_{jk}\cdot\vec{t}_k + \cdots + p_{jn}\cdot\vec{t}_n \tag{1}$$

其中, P_{jk}表示单个文档中不同主题的百分比; $\vec{t}_k$表示文档k对应主题的向量表示. 在模型迭代训练结束后, 融合文档权重向量和主题向量, 得到含有隐含主题信息的文档向量, 然后将词向量表示部分得到的枢轴词向量与该文档向量相加得到上下文向量, 以此来最小化主题预测过程中的负采样损失和 Dirichlet 似然项总和, 生成可解释的文档表示.

模型的总损失 L 是词向量表示部分的损失与文档向量表示部分的损失之和, 计算公式如式 (2) 所示:

$$L=\lambda\sum_{k=0}^{n}(\alpha-1)\log_2 p_{jk}+\log_2\sigma\left(\vec{c}_j\cdot\vec{w}_i+\sum_{l=0}^{n}\log_2\sigma\left(-\vec{c}_j\cdot\vec{w}_l\right)\right) \tag{2}$$

其中, $\vec{c}_j$表示上下文向量; $\vec{w}_i$表示目标词向量; $\vec{w}_l$表示词向量; λ表示超分布中的超参数; α表示先验参数. 通过调整λ, 发现在α<1 时, 主题的文档比例表现的比较稀疏, 其大多数值会接近于 0, 而在α>1 时, 主题的文档比例表现的更加密集, 为了增强模型的可解释性, 文中取$\alpha = n^{-1}$, n为主题数目, 迭代次数为 200, 通过不断对文本生成过程中的模型参数迭代优化, 最小化模型损失, 使得文档比例更密集, 词向量和主题的相似度更高, 主题的可解释性更强, 为后续热点话题聚类提供准确率更高、可解释性更好的主题表示.

2.4 文本相似度计算方法

文本相似度作为衡量不同文本间相关程度的指标, 是热点话题聚类过程中不可或缺的一部分. 文本间相似度越高, 说明其内容语义更接近. 在热点话题聚类任务中, 设计合理的相似度计算方法, 可以使聚类的性能更

优, 话题的凝聚度更好. 余弦相似度方法[18] 通过计算两个向量在向量空间方向上的余弦值, 来度量文本间相似度. 当两个向量属于同一方向时, 余弦值越接近 1, 两个向量就越相似, 表明该报道越可能聚类到该话题下. 利用余弦相似度计算向量集合$a=(a_1,a_2,\cdots,a_i,\cdots,a_n)$和$b=(b_1,b_2,\cdots,b_i,\cdots,b_n)$的语义相似度$sim(a,b)$的计算公式如式 (3) 所示:

$$sim(a,b)=\frac{\sum_{i=1}^{n}a_ib_i}{\sqrt{\sum_{i=1}^{n}a_i^2}\sqrt{\sum_{i=1}^{n}b_i^2}} \tag{3}$$

其中, a_i、b_i表示文本对应的主题特征词概率向量, 表示形式为(t_i,w_i), t_i表示特征词, w_i表示该特征词的权重.

新闻的实时增长性决定了一个话题结束后, 会继续出现新的话题. 利用文本表示模型提取主题特征词, 如果新话题存在很多与旧话题相同的特征词, 其文本相似度就会超过给定的相似度阈值, 此时就会将新的话题归到旧话题中, 这种情况下, 想要改善聚类质量, 就可以通过新的新闻报道发布的时间和旧话题中最先出现出现的新闻报道发布时间进行比较, 时间差越大, 不属于该话题的可能性就越大. 所以, 在话题生成的过程中, 考虑利用时间特性优化文本相似度算法, 用来更好的区别当前报道是否属于已有的话题, 提高聚类精度. 文中结合联合训练得到的热点主题特征词和时间特性, 将文本表示为$(t_i,w_i,(t_l,t_b))$, 其中t_i表示利用本文主题表示模型提取的隐含主题特征词, w_i表示对应特征词的权重, t_l表示话题特征词在对应话题报道中最后出现的更新时间, t_b表示该话题特征词在报道中第一次出现的时间. 在最新报道与已有文本出现相同特征词时, 其与相应新闻话题的时间差d_t的计算如式 (4) 所示:

$$d_t=t_l-t_n \tag{4}$$

其中, t_n表示该话题特征词在报道中最新出现的时间, 由于新闻报道随着时间差d_t的增大, 文本相似度会降低, 反之, d_t减小, 文本相似度会增大, 文中采用增函数的方式进行表示, 即: $f(x)=\frac{1}{1-x}$, 同时为了保证其在 (0, 1] 上连续变化, 文中令$x=t_n-t_l$, 时间相似度计算公式如式 (5) 所示:

$$sim_t=\frac{1}{1-(t_n-t_l)} \tag{5}$$

基于文中文本表示方法和余弦相似度, 得到报道的文本相似度算法公式如式 (6) 所示:

$$sim=sim_t\cdot sim(a,b) \tag{6}$$

采用式 (6) 计算文本语义相似度, 在对新增量的文本进行相似度计算时, 不需要重复计算与话题集合下的每篇新闻报道的相似度, 只需计算其对应文本表示向量与该话题中多篇报道特征向量平均值的相似度值, 这样不仅提升了文本相似度的计算效率, 节省了文本聚类时间, 也有效提升了聚类算法对新文本的反应能力.

2.5 SP-HTD 增量文本聚类

增量聚类主要是用来观察和发现动态数据流中文本信息的变化趋势. 与其他聚类算法不同的是, 在算法初始化时, 增量聚类不需要预先设定类簇的个数、初始中心点和结束条件, 在对新的文本数据加入时, 会依据一定的类簇划分规则形成新的类簇、或加入原有类簇、或造成原有类簇的分裂或合并, 在处理新数据时更便捷、高效, 能够提升话题聚类的效率.

假设文本的向量表示为$D=(d_1,d_2,\cdots,d_k,\cdots,d_n)$, 其中$d_k$表示第$k$个特征词对应的向量表示, D_0表示初始的文本聚类类簇, 对于动态增加的文本数据流, 具体识别规则如下: 在整个聚类过程中, 文本的初始类簇只有一个, 利用当前文本和已存在的类簇中心分别计算相似度, 判断新数据与最大相似度和阈值的关系, 如果大于阈值, 则归类到该类簇中, 否则添加新的类簇, 即标记新的增量节点, 以此动态增加类簇, 遍历至无输入新数据时, 算法结束, 完成文本的聚类. 可以看出, 对初始类簇的选择会对聚类结果产生很大的影响, 且对文本的相似度阈值比较敏感.

针对 Single-Pass 聚类算法不足, 考虑到热点话题检测任务的扩展性和性能需求, 本文做了以下改进: (1) 动态更新类簇中心, 通过文本发布时间和时间阈值不断优化, 避免重复的簇内相似度比较, 减小算法计算次数, 提高话题聚类的质量. (2) 对要聚类的文本数据按照话题的发布时间进行排序, 并采用 Single-Pass 算法对其进行初始化粗聚类, 然后将该聚类结果作为下一次文本聚类的输入来进行迭代聚类, 以此来降低聚类结果对文本输入顺序的过于敏感的问题. (3) 细化话题划分粒度, 选取文本表示模型提取的话题对应的主题词来划分子话题, 提升对报道间相似度计算的准确性. 算法流程如图 3 所示.

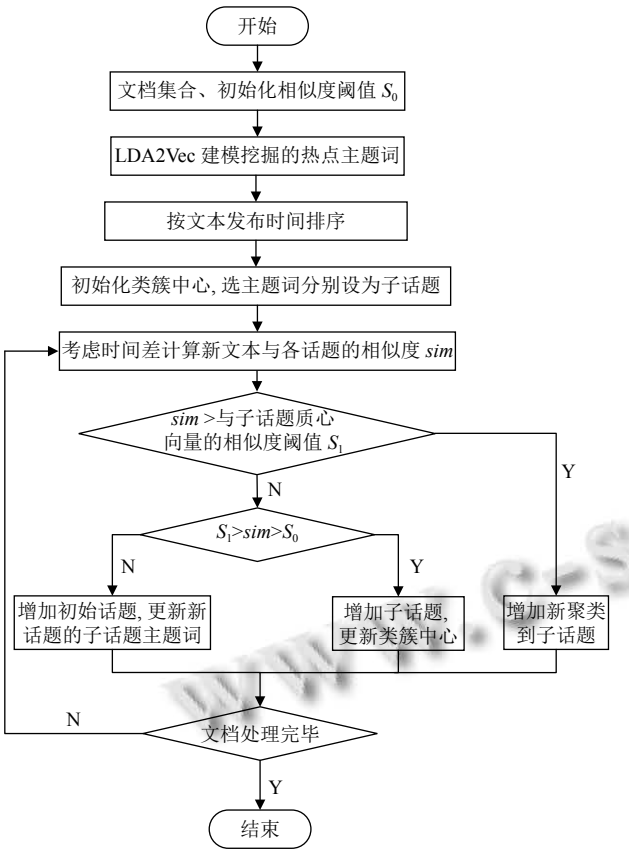

图 3 SP-HTD 算法流程图

具体算法步骤如下:

步骤 1. 输入初始文本集合和文本相似度阈值S_0;

步骤 2. 依据本文第 2.3 节的主题表示模型提取得到文本集合隐含的热点主题词;

步骤 3. 考虑其文本时间特性, 按照发布时间先后将文本数据集进行排序;

步骤 4. 选取步骤 3 中得到的当前输入文本对应话题的前 3 个热点主题词分别设为子话题, 然后初始化设定类簇中心D_0, 以此代表该聚类中所有文档具有的共同话题;

步骤 5. 依据本文第 2.4 节的文本相似度计算方法, 计算要新输入的文本与各子话题之间的相似度sim;

步骤 6. 判断如果计算的相似度值sim大于新文本与子话题的质心向量的相似度阈值S_1, 则增加新聚类到子话题, 否则执行步骤 7;

步骤 7. 考虑将计算的sim与S_0、S_1同时比较, 若处于两者之间, 则增加新的子话题, 同时更新类簇中心. 否则执行步骤 8;

步骤 8. 如果计算的相似度不在S_0、S_1之间, 则增加新的初始话题, 同时更新新话题对应子话题的 3 个主题词, 执行步骤 9;

步骤 9. 判断文本是否处理完毕, 如果处理完毕, 则结束聚类过程, 否则继续输入新文本, 从步骤 5 继续进行迭代聚类, 直至算法结束;

步骤 10. 输出 SP-HTD 聚类算法得到的热点话题聚类结果.

在处理输入的新文本时, 通过动态更新类簇中心, 仅仅需要将输入的新文本与该类簇的子话题质心向量比较相似度, 就可以判断是否属于该聚类, 减少了比较的次数, 降低了算法运算复杂度, 提高了新文本反应能力. 在子话题主题词选择时, 选择前 3 个主题词, 原因在于选取的主题词太多, 会增加后续输入文本与话题类簇中心相似度比较的时间, 选取的太少又会使得话题划分不够精细. 因此本文选择前 3 个主题词作为对应话题的子话题, 在计算文本相似度时保留更多新闻文本之间的相似性, 提高热点话题聚类的效率.

3 实验及结果分析

3.1 数据集与实验设置

本文通过爬取《纽约时报》《泰晤士报》等 10 个国际主流媒体近 10 年内有关中国的新闻报道作为语料库, 并将其分为经济、政治等 8 组不同类别的文档集. 在预处理阶段, 对数据进行降噪处理, 包括过滤停用词、去除重复文本数据和对缺失值进行正则匹配等操作, 最终获得 22 731 篇有效报道数据. 实验将词向量维度设置为 350 维, 初始率设为 0.06, 同时采用 GloVe 词向量模型[19] 初始化英文词向量, 获得数据集的全局共现信息. 具体数据组成如表 1 所列.

表 1 实验数据组成表

类别	数量 (篇)	类别	数量 (篇)
地理	1 160	经济	9 941
军事	1 850	科技	2 431
政治	1 890	社会	990
外交	2 842	文化	1 627

3.2 评价指标

本文采用热点话题检测常用的评价指标准确率P、召回率R和F值对话题检测的精度进行评估. 计算公式如下:

$$P = \frac{A}{A+B} \tag{7}$$

$$R = \frac{A}{A+C} \tag{8}$$

$$F = \frac{2PR}{P+R} \tag{9}$$

其中, A表示预测正确, 实际也正确的聚类元素数量, B表示预测正确, 实际不正确的聚类元素数量, C表示预测不正确, 实际正确的聚类元素数量. 可以看出F值越大, 说明话题检测的效果越好.

采用漏检率P_m (missing detection rate) 和误检率P_f (false detection rate) 对改进算法得到的聚类结果进行评测, 评估聚类效果[20]. 计算公式如下:

$$P_m = \frac{D_a}{D_a + D_b} \tag{10}$$

$$P_f = \frac{D_c}{D_c + D_d} \tag{11}$$

其中, P_m表示相关文档的漏检率. P_f表示不相关文档的误检率. D_a表示被检测到的相关文档数, D_b表示未检测到的相关文档数. D_c表示被检测到的不相关文档数, D_d表示未检测到的不相关文档数.

3.3 结果与分析

为了评估本文 SP-HTD 聚类算法的聚类结果的可行性和有效性, 在第 2.3 节主题模型对数据集进行文本表示的基础上, 以 Single-Pass (SP) 聚类算法、文献 [21] 提出的 SP-NN 和 SP-WC 聚类算法为基线, 将 4 种算法在测试集上进行话题聚类任务, 其结果如图 4 所示.

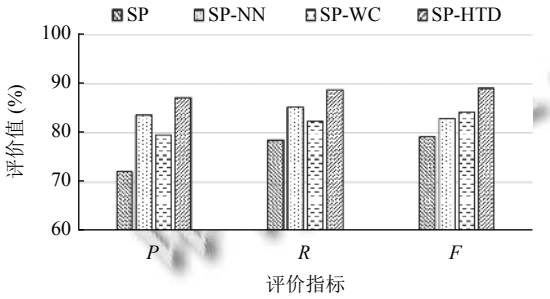

图 4 P、R 和 F 值结果比较

从图 4 可以看出, 在热点话题检测任务上, SP-HTD 聚类算法的F值最高可达 89.3%, 相比于 SP、SP-NN 和 SP-WC 在准确率分别提高了 15%、3.6%、7.5%, 在召回率上分别提高了 10.2%、3.5%、6.3%, 均有更好的效果, 表明 SP-HTD 聚类算法能够将文本聚类到更好的话题类别, 热点话题聚类效果更好. 原因在于本文算法考虑了更全面的语义特征信息, 联合训练文档向量和词向量, 挖掘的主题表示更为精确, 并且在文本相似度计算时, 考虑了新闻报道的时效性, 通过报道发布的时间差, 动态更新质心向量, 提高了热点话题聚类的准确率.

本文采用漏检率和误检率对话题聚类结果的质量进行对比评估, 从数据集中选取 6 个热点话题, 按 8:2 的比例选取每个话题的文本作为聚类训练数据集和验证数据集, 将其经过文本表示模型的建模后作为聚类算法的输入, 采用 SP、SP-NN、SP-WC 和 SP-HTD 聚类算法分别进行实验, 其结果如图 5、图 6 所示.

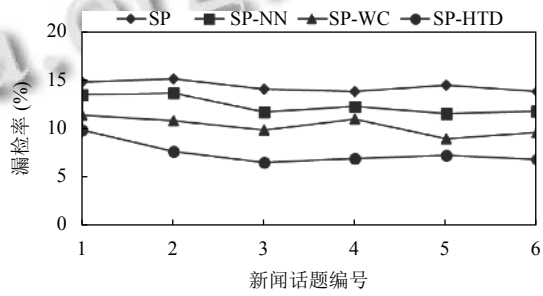

图 5 漏检率比较

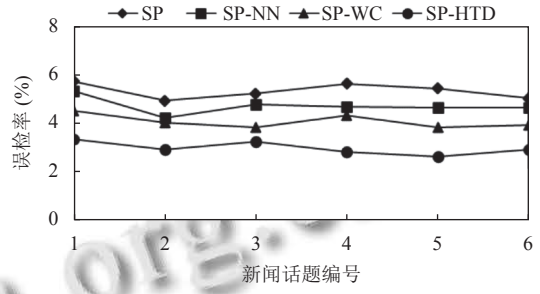

图 6 误检率比较

从图 5、图 6 可以看出, 对相同的新闻数据集进行热点话题检测的话题聚类任务, 文中提出的 SP-HTD 聚类算法相比于 SP、SP-NN 和 SP-WC 聚类算法得到的漏检率分别可降低约 7.6%、6.1%、4.1%, 误检率可降低约 3.1%、2.3%、1.5%. 其中, 与 SP-WC 算法相比, 话题 1 和话题 5 的漏检率差距较小, 话题 2 和话题 4 的漏检率差距较大. 与 SP 算法相比, 在话题 4 和话题 5 的误检率差距较大, 话题 2 和话题 3 的误差率差异较小, 但综合来看, 本文提出的 SP-HTD 聚类算法提高了话题检测聚类的质量. 原因在于本文算法在处理新文本时, 无需重复计算整个文档集, 并且根据时间阈值, 在聚类过程中不断优化类簇中心, 保证了聚类算法对新文本扩展性能和聚类质量.

4 结束语

本文提出了一种面向热点话题检测任务的增量文本聚类算法 (SP-HTD), 针对 Single-Pass 算法对数据的输入顺序过于敏感和聚类效率相对低的问题, 从文本表示、相似度计算和文本聚类 3 个方面进行了改善, 并与 SP、SP-NN 和 SP-WC 聚类算法做对比实验. 结果表明, 在热点话题检测任务上, 相比传统的 Single-Pass 算法, 在保证聚类精度的前提下, 所提算法计算的聚类中心的代表性更强, 可以有效提高话题检测的准确性. 在下一阶段工作中, 将考虑更进一步细化话题检测粒度, 对特定话题下的子话题, 研究其内部结构和联系, 以期实现更好的热点话题检测效果.

参考文献

1 Miao ZC, Chen K, Fang Y, *et al*. Cost-effective online trending topic detection and popularity prediction in microblogging. ACM Transactions on Information Systems, 2017, 35(3): 18.

2 Mamo N, Azzopardi J, Layfield C. An automatic participant detection framework for event tracking on Twitter. Algorithms, 2021, 14(3): 92. [doi: 10.3390/a14030092]

3 周楠, 杜攀, 靳小龙, 等. 面向舆情事件的子话题标签生成模型 ET-TAG. 计算机学报, 2018, 41(7): 1490–1503. [doi: 10.11897/SP.J.1016.2018.01490]

4 Indra, Winarko E, Pulungan R. Trending topics detection of Indonesian tweets using BN-grams and Doc-p. Journal of King Saud University—Computer and Information Sciences, 2019, 31(2): 266–274. [doi: 10.1016/j.jksuci.2018.01.005]

5 Moody CE. Mixing Dirichlet topic models and word embeddings to make LDA2Vec. arXiv: 1605.02019, 2016.

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

7 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. arXiv: 1301.3781, 2013.

8 Gupta C, Grossman R. Genic: A single pass generalized incremental algorithm for clustering. Proceedings of 2004 SIAM International Conference on Data Mining. Orlando: Society for Industrial and Applied Mathematic, 2004. 147–153.

9 Allan J, Papka R, Lavrenko V. On-line new event detection and tracking. Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Melbourne: ACM, 1998. 37–45.

10 黄建一, 李建江, 王铮, 等. 基于上下文相似度矩阵的 Single-Pass 短文本聚类. 计算机科学, 2019, 46(4): 50–56. [doi: 10.11896/j.issn.1002-137X.2019.04.008]

11 张帆, 潘亚雄, 胡勇. 基于改进 Single-Pass 的新闻话题检测与追踪技术研究. 信息安全研究, 2020, 6(5): 396–403.

12 武森, 高晓楠, 何慧霞. 基于双向改进余弦相似度的话题发现算法. 运筹与管理, 2021, 30(2): 75–83.

13 Sharma D, Kumar B, Chand S. A survey on journey of topic modeling techniques from SVD to deep learning. International Journal of Modern Education and Computer Science (IJMECS), 2017, 9(7): 50–62. [doi: 10.5815/ijmecs.2017.07.06]

14 Li SH, Chua TS, Zhu J, *et al*. Generative topic embedding: A continuous representation of documents. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: ACL, 2016. 666–675.

15 余传明, 原赛, 朱星宇, 等. 基于深度学习的热点事件主题表示研究. 数据分析与知识发现, 2020, 4(4): 1–14.

16 薛涛, 郭莹, 胡伟华. 基于 LDA2Vec 联合训练的热点主题识别方法. 西安工程大学学报, 2021, 35(4): 95–101.

17 Mikolov T, Sutskever I, Chen K, *et al*. Distributed representations of words and phrases and their compositionality. Proceedings of the 26th International Conference on Neural Information Processing Systems. Lake Tahoe: Curran Associates Inc., 2013. 3111–3119.

18 Akbaş CE, Günay O, Taşdemir K, *et al*. Energy efficient cosine similarity measures according to a convex cost function. Signal, Image and Video Processing, 2017, 11(2): 349–356. [doi: 10.1007/s11760-016-0949-7]

19 Pennington J, Socher R, Manning CD. GloVe: Global vectors for word representation. Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing. Doha: ACL, 2014. 1532–1543.

20 理姗姗, 杨文忠, 王婷, 等. 基于网络社交媒体的子话题检测技术综述. 计算机应用, 2020, 40(6): 1565–1573.

21 Karypis G, Han EH, Kumar V. Chameleon: Hierarchical clustering using dynamic modeling. Computer, 1999, 32(8): 68–75. [doi: 10.1109/2.781637]

(校对责编: 孙君艳)