中国科学院大学
University of Chinese Academy of Sciences

硕士学位论文

面向大规模网络数据的话题检测研究

作者姓名：林尽忠

指导教师：刘玉贵 副教授

中国科学院大学计算机科学与技术学院

学位类别：工程硕士

学科专业：计算机技术

培养单位：中国科学院大学计算机科学与技术学院

2019 年 6 月

Topic Detection Research for Large-Scale Web Data

A thesis submitted to the
University of Chinese Academy of Sciences
in partial fulfillment of the requirement
for the degree of
Master of Computer Technology
in Computer Science and Technology

By

Jinzhong Lin

Supervisor: Professor Yugui Liu

School of Computer Science and Technology,
University of Chinese Academy of Sciences

June, 2019

中国科学院大学
学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日　　期：

中国科学院大学
学位论文授权使用声明

本人完全了解并同意遵守中国科学院大学有关保存和使用学位论文的规定，即中国科学院大学有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延迟期后适用本声明。

作者签名：　　导师签名：

日　　期：　　日　　期：

摘 要

随着信息技术和移动网络技术的快速发展，人们能够越来越方便地通过网络在社交媒体上获取信息和交换意见。因此，极大地促进了用户生成式内容的产生和传播。但是，海量的数据使得用户难以从中快速有效地提取当前热点话题以及感兴趣的话题。本文主要对当前网络话题检测在大规模网络数据上的可扩展性进行研究，在三个方面加以改进：网络话题生成，网络话题质量排序，并行化处理。

首先，本文研究了网络话题的生成任务。我们用相似度图表示网页之间的关系。由于网络中含有大量噪声网页，所以我们通过一定的阈值截断该相似度图并只保留最相关的一定个数的近邻网页间的相似度值。。。。

其次，本文研究了网络话题的质量排序。

最后，本文就质量排序的过程，进行了并行化的处理。

关键词：网络话题检测，大规模网络数据，泊松去卷积算法

Abstract

Abstract:

Keywords: Topic Detection on Web, Large-Scale Web Data, Poisson Deconvolution

目 录

图形列表

表格列表

符号列表

字符

Symbol	Description	Unit
R	the gas constant	$m^2 \cdot s^{-2} \cdot K^{-1}$
C_v	specific heat capacity at constant volume	$m^2 \cdot s^{-2} \cdot K^{-1}$
C_p	specific heat capacity at constant pressure	$m^2 \cdot s^{-2} \cdot K^{-1}$
E	specific total energy	$m^2 \cdot s^{-2}$
e	specific internal energy	$m^2 \cdot s^{-2}$
h_T	specific total enthalpy	$m^2 \cdot s^{-2}$
h	specific enthalpy	$m^2 \cdot s^{-2}$
k	thermal conductivity	$kg \cdot m \cdot s^{-3} \cdot K^{-1}$
S_{ij}	deviatoric stress tensor	$kg \cdot m^{-1} \cdot s^{-2}$
τ_{ij}	viscous stress tensor	$kg \cdot m^{-1} \cdot s^{-2}$
δ_{ij}	Kronecker tensor	1
I_{ij}	identity tensor	1

算子

Symbol	Description
Δ	difference
∇	gradient operator
$\delta^{\pm}$	upwind-biased interpolation scheme

缩写

CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewy
EOS	Equation of State
JWL	Jones-Wilkins-Lee
WENO	Weighted Essentially Non-oscillatory
ZND	Zel'dovich-von Neumann-Doering

第 1 章　绪论

1.1　课题研究背景

1.1.1　课题背景与意义

1.1.2　研究问题与难点

1.2　国内外研究现状

1.3　常用数据集

在话题检测领域，MCG-WEBV[xxx] 和 YKS[xxx] 是两个常用的数据集。许多网络话题检测算法使用这两个数据集来验证算法的性能。表1.1汇总了这两个数据集的基本信息。

表 1.1 数据集基本情况汇总

数据集	话题数量	网页数量	所有话题包含网页数	词典规模	平均每个网页包含词语数量
MCG-WEBV	73	3660	832	9212	35
YKS	298	8660	990	80294	228

1.3.1　MCG-WEBV

MCG-WEBV 数据集爬取了 YouTube 自 2008 年 12 月到 2009 年 2 月间的“浏览最多”的视频，包含了 15 类。同时还爬取了与这些视频相关的视频以及相同作者上传的视频。最终 MCG-WEBV 包含了 80031 个视频。

除了视频数据外，MCG-WEBV 还包含丰富的信息：

- 5 种元特征：视频 ID、上传用户名、上传时间、视频长度、视频类别；
- 人工分类的 15 类视频类别及其标签；
- 8 种网页特征：标题、描述、标注、评级、评论数、拍摄张数等；
- 9 种视觉特征：166 维颜色直方图特征、320 维边缘直方图特征等；
- 采用文本特征模型产生的文本特征和 36 维的音频特征；

MCG-WEBV 对核心数据集进行了人工标注，最终得到 73 个话题。这些话题由话题热度决定。而话题热度主要与话题关注度和持续时间相关。在 MCG-WEBV 中，话题的关注度由话题包含的视频数和视频点击数决定，话题的持续

时间为话题中第一个视频的上传时间与用户最后观看时间之间的间隔。所以，话题的热度由公式1.1决定。其中 $\tau(t)$ 指话题 t 的持续时间，$N(t)$ 表示话题 t 的视频总数，$V(t)$ 表示话题 t 中视频被观看的次数。

$$H(t) = \log \frac{|N(t)| * |V(t)|}{\tau(t)} \tag{1.1}$$

对于热门话题的标注，首先通过算法对提取的文本信息使用自适应的 k 均值算法聚类，得到 113 个话题。然后通过人工筛选，剔除持续时间短、网页少的的话题，删除话题内无关的网页，融合语义相近的话题，最终得到 73 个人工标注的网络话题。

1.3.2 YKS

YKS 是一个跨媒体的多模态数据集。主要由优酷网的视频数据和新浪网的新闻数据共同组成。其中，约有 75% 的数据只包含文本信息，约有 25% 的数据同时包含文本信息和视觉信息。此外还有极少量的数据只包含视觉信息。

对于优酷视频，逐日爬取了从 2012 年 5 月 1 号开始的视频点击率在 5 万以上的视频，总共得到 5500 多个视频。然后经过过滤、剔除长度小于 5 秒和大于 1 小时的视频，最终得到 2131 个视频数据。除了视频外，还有其他相关的数据，比如视频标题、视频标注、视频描述、视频点击率、上传时间等。

对于新浪网的新闻数据，爬取了从 2012 年 5 月 1 号到 2012 年 5 月 31 号的新浪网发布的所有新闻。包括新闻标题、新闻正文（文本、图像、视频）和其他辅助信息，比如新闻发布时间、新闻标签、新闻点击率、新闻相关链接等。总共有 30000 多篇新闻文档，经过过滤空白新闻、纯图像新闻、纯视频新闻等处理后，最终剩余 7325 篇新闻文档。

YKS 数据集同样经过了话题的人工标注。总共标注了 318 个话题，其中 225 个只包含新浪新闻的纯新闻话题，20 个只包含优酷网的纯视频话题，另外 73 个话题同时包含新浪新闻和优酷视频。

1.4 论文内容与组织结构

第 2 章　网络话题的快速生成

2.1　引言

随着社交媒体的快速发展，越来越多的用户通过社交媒体来获取信息和分享观点。由此产生了海量的用户生成式数据，使得用户难以从中快速获取热点话题及感兴趣的话题。话题是某个种子事件及其相关报道的集合。网络话题检测任务自动地将网络数据组织成更多有意义的热门话题。从本质上来讲，网络话题检测就像从大海里捞针，类比从大量的网络数据中找到一小部分感兴趣数据并将其组织成热门事件。

传统的话题检测任务致力于将每个新闻文章分配到至少一个话题中。而这些经过专业编辑的新闻文章数据与网络数据存在极大的差异：由于社交媒体对所发布内容的约束较少，所以来自社交媒体的网络数据更加简短、稀疏并且充满噪声；在大量网页中，只有一小部分的网页能够被组织成热点话题。因此网络话题检测不仅面临着低效的特征表达还需要处理大量的噪声网页。

网络话题检测的关键问题是如何在海量噪声网页存在的前提下组织热点话题。一个直观的方法是在噪声网页存在的情况下去聚类网络话题。然而海量的噪声网页使得传统方法不再适用。为了移除大量噪声网页带来的不利影响，传统的方法采用了一种看似合理的假设，即在相同话题下的任意两个网页之间的相似度应该大于该话题中的网页和噪声网页之间的相似度。然而，这个假设也很难站得住脚，主要存在如下两个挑战：

1) 稀疏和充满噪声的网络数据导致低效的特征：用户生成式数据几乎没有约束，所以传统的适用于长文本的 TF-IDF 特征不足以高效表达社交媒体上稀疏的、充满噪声的网络数据。

2) 低阶特征和高阶语义间存在的语义鸿沟：低阶的特征难以准确地表达高阶语义间的关系。所以网页之间更大的相似度值并不意味着这两个网页在语义上更加相似。

在低效的特征表示以及海量噪声存在的前提下，我们寻找一种无关模型、无关优化的的方法来生成网络话题。首先是因为网络话题的结构和内容差异性很大，一个无模型的方法能拥有好的生成话题能力；其次，为了避免高复杂度的优化措施，一个无需优化的方法能够更好地处理大规模数据问题；然后，我们避免去处理短文本如何编码生成高效的特征表示这样一个开放性问题；最后网络话

题面临着海量噪声。

我们研究了网络话题在相似度空间上的统计模式，发现同一个话题下的所有网页之间的相似度与 Lévy Walks 存在统计意义上的相似性。具体地，我们将网页之间的相似度类比为 Lévy Walks 中的步长，那么一个热点话题中的所有相似度分布大致服从重尾分布，而这重尾分布又是 Lévy Walks 中步长的特性。Lévy Walks 是一种随机游走模型，其步长服从重尾概率分布。一次移动定义为一个质点从一个位置无偏移的移动到另一个位置的步长。直观上讲，Lévy Walks 包含许多短步长的移动和一些逃脱短步长控制的额外较长步长的移动。因此，Lévy Walks 可以用来很好地描述觅食动物的迁徙模式。

当 Lévy Walks 被用来组织网络话题，关键问题变成如下几点：

1) 如何在未知参数下模拟 Lévy Walks 中额外长的步长。

2) 如何确定话题所需要的步长个数即网页个数。

3) 在组织话题的时候选择这个步长是否能带来好处。

基于上述三个问题，我们提出以下解决方案：

1) 提出基于 Lévy Walks 的话题生成算法（Lévy Walks-based Topic Generation，LWTG），该方法通过模拟 Lévy Walks 的特性，采用基于种子网页的多分配策略，优雅地避免了不同 Lévy Walks 需要不同参数的麻烦。简单的同时速度也很快。

2) 提出多阈值方法来截断网络话题的生长，这会带来一系列过完备话题，从而提高话题的召回率。

3) 至于话题召回准确率是由后续的排序算法来保证的。

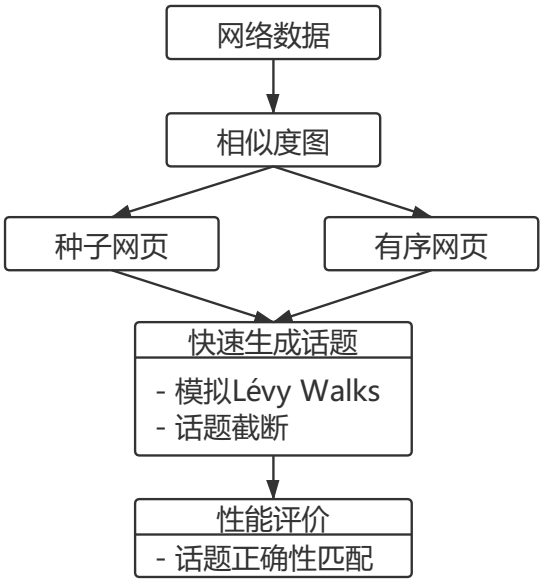


图 2.1 通过模拟 Lévy Walks 来快速生成网络话题的流程图

本论文是第一个发现网络话题检测在相似度空间上和 Lévy Walks 具有相似的特点，并且进行了一系列实验来阐述这个发现所带来的好处。我们提出的 LWTG 方法不仅简单快速，而且能够进一步提高话题检测的召回率。通过简单地对网页进行分配，无需参数优化，我们找到了一种新的组织网络话题的方法。我们的方法在网络话题检测效率方面已经远超当前最好的方法，而且在话题检测召回率方面也能够赶上甚至超越当前最好的方法。算法框架如图2.1所示。

2.2 相似度图的构建

本章的研究重点是网络话题的生成，因此，为了减少其他因素的影响，我们忽略网络数据的多模态性和时间戳、链接等其他信息，只使用纯文本信息。所以每个网页就是一个文本字符串。对于给定的网络数据集，包含一系列的网页 $W = \{w_1, ..., w_N\}$，我们对其进行处理，生成一个 knn 近邻的相似度图 $G = (V, E, A)$。其中顶点集 V 对应网页集合，仿射矩阵 A 对应截断后任意两个网页之间的相似度，边集 E 对应任意两个网页之间的非 0 边 [xx]。具体处理如下。

首先，我们对网页的文本字符串进行分词，然后使用词袋模型 [xx] 对网页文本进行基本的统计和表示，再用 TF-IDF 特征值表示每个词的权重，这样每个网页就可以用一个特征向量 $x_i \in \mathbb{R}^M (i = 1, ..., N)$ 表示。其中 M 是表示词典大小，N 表示网络数据集中网页的数量。对于 TF-IDF 特征，TF 表示词频，IDF 表示逆文档频率。一个词在文档中出现的频率越高，其词频值越大，相应 TF-IDF 值就越大。与此同时，一个词如果出现在越多的文档中，则其逆文档频率值越低，相应 TF-IDF 特征值就越低。逆文档频率降低那些高频出现但是较没有判别意义的词的权重。比如：‘我’、‘那’这种指示代词几乎没有判别能力，虽然它们的词频值很大，但是它们的逆文档频率值很低，进而使得 TF-IDF 值很低。虽然还可以使用更高级的特征，但是语义鸿沟问题仍然存在，而且使用 TF-IDF 特征足够简单，能够带来更快的处理速度。

然后，基于得到的每个网页的特征向量，我们可以通过余弦距离来度量两个网页之间的相似度大小。在公式2.1中，S_{ij} 表示两个网页之间的相似度，x_i 和 x_j 表示两个网页的特征向量：

$$S_{ij} = \frac{x_i \cdot x_j}{|x_i||x_j|} \tag{2.1}$$

最后，得到相似度图后，在每个网页中，保留与其语义最相近的 knn 个网页关系，删除其他网页关系。这里假设两个网页之间相似度越高则语义越相近。这样做能够过滤掉大量网页之间的噪声干扰，即不相关的噪声网页。网络数据集中

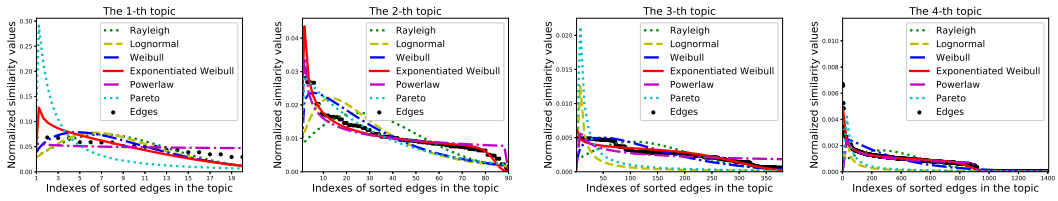
的噪声网页越多，knn 应该选择更低的值。在公式2.2中 $KNN(w_i)$ 表示与网页 w_i 语义最相近的 knn 个网页集合。同时，我们使用公式2.3将两个网页之间低于某个阈值的相似度置为 0，因为我们认为过低的相似度表示这两个网页在语义上已经没有关联关系了。这进一步过滤了噪声网页。

$$S_{ij} = \begin{cases} 0, & w_j \notin KNN(w_i) \\ S_{ij}, & w_j \in KNN(w_i) \end{cases} \tag{2.2}$$

$$S_{ij} = \begin{cases} 0, & S_{ij} < \epsilon \\ a_{ij}, & S_{ij} \geq \epsilon \end{cases} \tag{2.3}$$

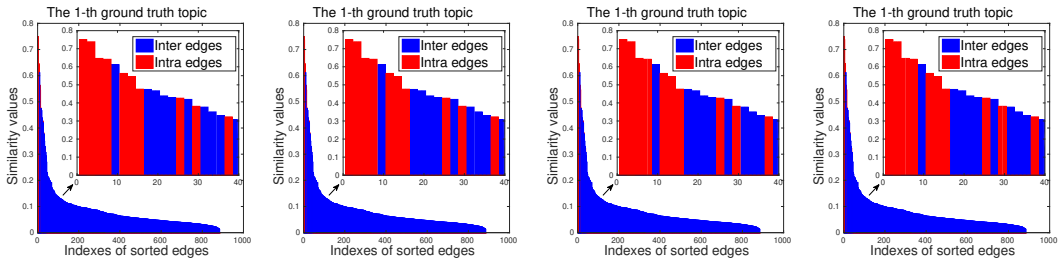
2.3 网络话题的 Lévy Walks 特性

一个网络话题由未知数量的网页构成，在相似度空间上表示为一系列边 e_{ij} 经过仿射后的相似度 a_{ij}，其中网页 x_i 和 x_j 属于该话题。话题的相似度空间包含两种相似度：1）话题内部相似度边：同个话题内部两个网页之间的相似度；2）话题之间相似度边：两个不在同个话题内的网页之间的相似度。



(a) **(b)** **(c)** **(d)**

图 2.2 第 1、2、3、4 个话题分别最拟合指数韦伯分布，幂次分布，指数韦伯分布，幂次分布。



(a) **(b)** **(c)** **(d)**

图 2.3 已排序边的分布情况，包含话题内部的边和话题之间的边。

表 2.1 常见的重尾分布函数。

分布函数	概率密度函数
指数韦伯分布 [1]	$(1-\exp(-\frac{x}{\lambda})^k)^\alpha$
锐利分布	$\frac{x}{\sigma^2}\exp(\frac{-x^2}{2\sigma^2})$
韦伯分布 [2]	$\frac{k}{\lambda}(\frac{x}{\lambda})^{k-1}\exp(-(\frac{x}{\lambda})^k)$
对数正态分布	$\frac{1}{x\sigma\sqrt{2\pi}}\exp(-\frac{\ln(x)-\mu}{2\sigma^2})$
幂次分布	$a(cx)^{-k}$
帕累托分布 [3]	$\frac{\alpha a^\alpha}{x^{\alpha+1}}$

[1] $\alpha \geq 1$.

[2] $k < 1$.

[3] $0 < a \leq x$

我们从 MCG-WEBV 数据集中随机选择四个真实的网络话题来描述网络话题在相似度空间上的模式。对同个话题内部的相似度进行从大到小排序。图2.2说明同个话题下的已排序的相似度服从重尾分布。从图2.2引出两个问题：

1) 是否所有的话题服从相同的重尾分布?

2) 服从相同分布的话题是否拥有相同的分布参数?

为了解决上述问题，我们使用极大似然估计去将已排序的相似度拟合为已知的分布。例如：指数韦伯分布、锐利分布、韦伯分布、对数正态分布、幂次分布、帕累托分布。表2.1列出这些分布的概率密度函数。而且，为了量化最好的分布，我们引入赤池信息准则 AIC（Akaike’s Information Criterion）[xxx]：

$$AIC = -2log(L(\hat{\theta}|data)) + 2K \tag{2.4}$$

其中 $L(\cdot)$ 是似然函数，K 是参数的数量。由于 AIC 值容易受样本大小影响，不能被直接用来作为绝对的度量标准。所以，使用下面的转换形式作为每个模型的置信权重：

$$w_i = \frac{exp(-\Delta_i/2)}{\sum_{r=1}^{R} exp(-\Delta_r/2)} \tag{2.5}$$

其中 R 是分布函数数量。赤池信息权重被认为是分布函数可能性的归一化值。

从图2.2到2.3，我们可以得到下面观察结果：

1) 同个话题下网页之间的已排序的相似度与 Lévy Walks 存在统计意义上的相似特性。图2.2a、2.2b、2.2c、2.2d表明同个话题内的相似度服从重尾分布。

2) 不同的话题服从不同的重尾分布。例如,根据赤池信息权重公式2.5,图2.2a所表示的第一个话题服从指数韦伯分布，而图2.2b所表示的第二个话题服从幂次分布。

3) 热点话题中包含一些额外的边。如图2.3所示，如果话题的相似度按照递减排序，那么由边连接排在较前的网页，并不意味其绝对属于该话题。

如果将相似度类比为网页之间的步长，那么与 Lévy Walks 相比，网络话题在相似度空间上有两个统计意义上的相似特性：1）Lévy Walks 和网络话题的步长均服从重尾分布；2）Lévy Walks 和网络话题均包含许多短的步长 (高相似度) 和一些额外长（低相似度）的步长。至此，这两个相似特性被认为是网络话题的 Lévy Walks 特性。

2.4 通过模拟 Lévy Walks 生成话题

既然网络话题有 Lévy Walks 特性，我们试图从这个特性入手，从海量网络数据中生成话题。一个最简单的办法是根据重尾分布来组织网页进入对应话题。然而，正如之前所言，我们不可能提前训练一个通用的含有重尾分布函数的模型。

通过在相似度空间利用重尾分布的特点，我们认为可以在将网页分配给话题的时候适当添加一定的随机性以模拟 Lévy Walks 中额外较长的步长。为此，我们设计了一种通过模拟 Lévy Walks 来生成话题的算法（Lévy Walks-based Topic Generation，LWTG）。LWTG 算法的框架如图2.4所示。

图 2.4 LWTG 算法生成话题的框架图

本论文中，我们提出两种度量网页和话题间相似度的方法，并将网页分配给相似度最高的 k 个话题来模拟网络话题在相似度空间中的 Lévy Walks 特性：一些额外较低的相似度。

2.4.1 寻找种子网页

网络话题由初始核心事件不断在社交媒体上传播得以发展壮大。传播过程会逐渐吸收许多直接或者间接的外延信息。网络话题的形成可以被看做是一种

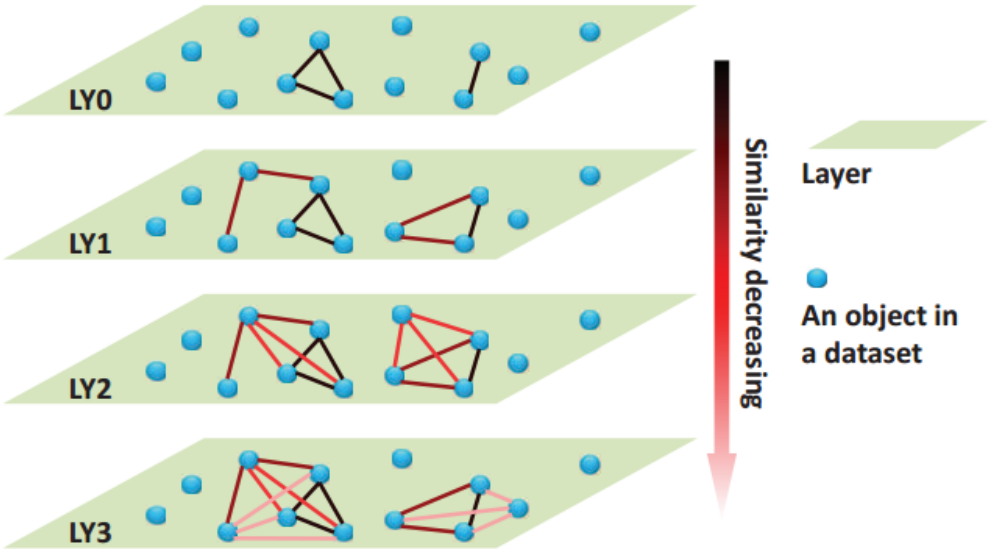


图 2.5 基于相似度流的网络话题演化示意图

信息扩散的过程。而信息扩散的过程肯定会有一定的损失。我们使用相似度流来模拟这种信息扩散过程，扩散过程中信息的损失对应于相似度流中相似度值的较小。图2.5展示了网络话题基于相似度流的演化过程：话题初始时只有一两个网页，随后在较低的相似度上吸收更多的网页来形成更大的话题。随着演化的进行，吸收的网页的相似度越来越低。这种过程我们称之为相似度流扩散（Similarity Cascade, SC）。又因为不同用户之间的需求是不一样的，导致用户对话题的理解具有极大的差异。所以对同一个核心事件演化形成的话题，不同的用户所理解的话题的规模也是不一样的。

既然话题是由核心事件演化来的，那么我们希望能够通过代表该核心事件的种子网页，进而通过相似度流的扩散过程来模拟演化过程，最终生成话题。首要问题是如何判断一个网页能否作为种子网页？直观上理解，在一个由多个网页构成的话题中，其网页分布应该是尽可能均匀且紧凑的。均匀表示该话题内的网页之间的相似度较为接近，这是因为话题内的网页都是在为该话题服务的，所以它们应该是相似的，即相似度应该尽可能一致。紧凑表示该话题内的网页与话题应该是紧密相关，即相似度应该尽可能大。

受到论文 [xxx] 的启发，我们引进了站点熵率（Site Entropy Rate，SER）用来度量网页成为种子网页的概率。通过将相似度流从一个网页转移到另一个网页的过程模拟为全连接图中从一个站点转移到另一个站点的随机游走的过程，SER意味着从一个网页在一步内转移到其他网页的平均总信息转移量。而由种子网

页吸收相似网页演化生成话题的过程中，越接近初始核心事件的网页，其通过相似度能够转移的平均总信息量也越大。SER 的公式如下：

$$SER_i = \pi_i \sum_{j \in <i>} -P_{ij} \log P_{ij} \tag{2.6}$$

其中 $P_{ij} = \frac{S_{ij}}{\sum_{j \in <i>} S_{ij}}$ 表示网页 w_i 转移到网页 w_j 的转移概率。$< i > \subset [1 : N]$ 保存了 s 个与网页 w_i 最相似的网页索引。公式2.6表明 SER 可以被分为两个部分：稳态分布项和熵项。这两部分作用分别如下：

1) 稳态分布项：$\pi_i = \frac{S_i}{S}$，其中 $S_i = \sum_{j \in <i>} S_{ij}$ 是从网页 w_i 出发的所有相关相似度的和，$S = \sum_i \sum_{j \in <i>} S_{ij}$ 是相似度图中的所有网页及其最相似的 s 个网页的相似度的和。π_i 被认为是网页 w_i 访问其他网页的频率；π_i 越大，则表示由网页 w_i 经过一步演化的话题更加的紧凑；

2) 熵项：$\sum_{j \in <i>} -P_{ij} \log P_{ij}$ 度量了网页 w_i 在一步内访问其他网页的不确定性。熵项越大，表明与网页 w_i 直接相连的其他网页的相似度分布更加均匀。

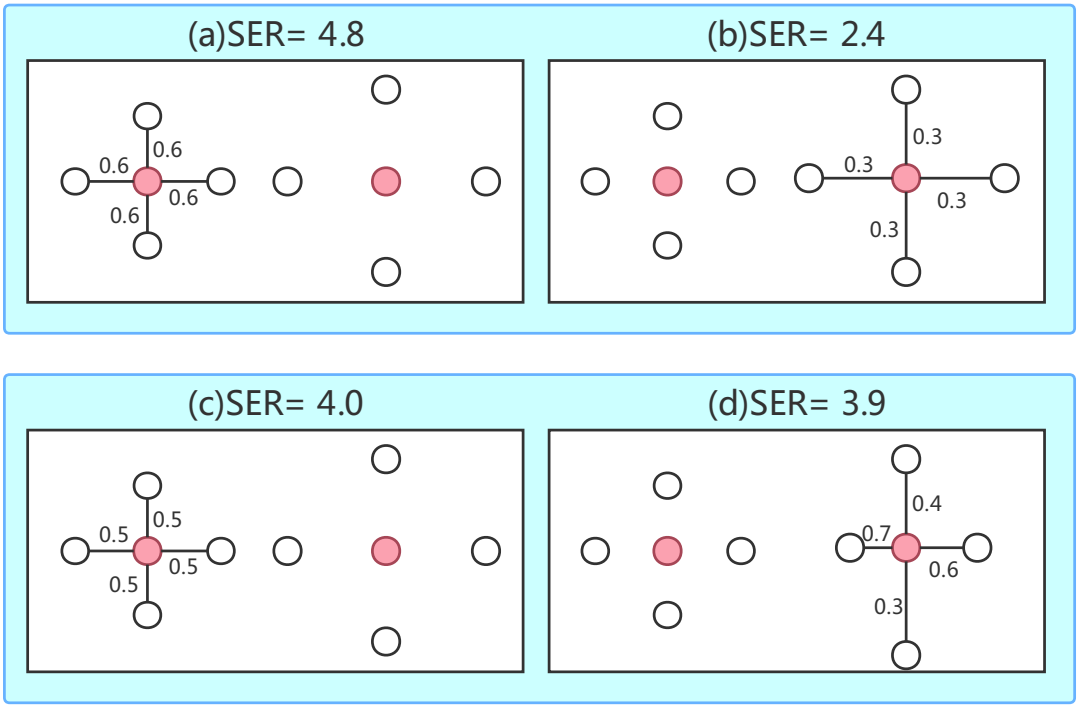


图 2.6 网页紧凑性和均匀性的 SER 评估

SER 由这两项的乘积构成。网页的 SER 越大，则表示其越有可能成为话题中的种子网页。如图2.6所示，图中空心圈圈表示网页，红色实心圆圈表示要评估的网页，数字表示网页之间的相似度。从图（a）和图（b）可以看出在相同均匀分布的情况下（熵项一致），越紧凑的网页有更大的 SER 值。从图（c）和图

(d) 可以看出在相同紧凑性情况下（稳态分布项 π 一致），相似度分布越均匀的网页有更大的 SER 值。

在定义了 SER 作为网页成为种子网页的概率情况下，我们提出了一种基于关联度过滤的贪心算法来找到不同规模的种子网页。算法1展示整个搜索过程。算法的输入为相似度图，然后根据相似度图来计算每个网页的 SER 值。这里我们只需要选择与网页最相似的 s 个网页的相似度来计算 SER 值即可。因为 SER 仅用作一步内的相似度转移，而在一步内转移的网页应该是比较相似且数量比较少的，而不是全部的网页。所以这里我们只选择了 s 个最相似的网页来计算 SER 的值。另一个关联度参数 d 用来确定种子网页后，过滤与其最相似的 d 个网页。这么做是因为我们认为这 d 个最相似的网页可以在之后由该种子网页生长而得到。d 值越大，产生的种子网页数量就越少。由于 d 值不好确定，同时为了更好地提高话题的召回率，我们使用多个不同的 d 值来产生多种规模的种子网页。算法输出为规模较小的种子网页集合。整个算法是基于 SER 值排序的贪心过滤算法：选择当前最优的种子网页，过滤与其最相似的 d 个网页，直到所有网页遍历结束。

算法 1 基于关联度的种子网页贪心搜索算法

Input: 相似度图 $G = (V, E, A)$，最相似网页个数 s，关联度 d

Output: 种子网页集合 SW，有序网页索引| $SortedIndex$

1: 初始化最终要生成的种子网页集合 SW 为空集
2: 根据公式2.6计算所有网页的 SER
3: 根据 SER 进行从大到小排序，得到有序且有连接的网页索引| $SortedIndex$
4: 找出与每个网页 w_i 最相似的 d 个网页索引, 记为 $\{i\}$
5: **for** $i \in SortedIndex$ **do**
6: **if** 网页 w_i 及其最相似的 d 个网页 $w_{\{i\}}$ 未被访问 **then**
7: $SW \leftarrow SW \bigcup w_i$
8: 标记网页 w_i 及其最相似的 d 个网页 $w_{\{i\}}$ 的状态为已访问
9: **end if**
10: **end for**

2.4.2 网页多分配算法

在产生种子网页后，我们提出了网页多分配算法来实现种子网页吸收相似网页，进而演化成话题。我们通过将网页分配给相似度最高的 k 个话题来模拟网

络话题 Lévy Walks 特性中额外较低的相似度。同时这个分配过程模拟了相似度流的扩散过程，所以我们采用了生成种子网页过程中产生的网页遍历顺序作为我们遍历网页的顺序。然后逐个进行网页分配。算法2展示了这个分配过程。其中对于每个待分配的网页，我们需要计算其与种子话题的相似度，希望这个相似度能够反映种子话题对网页的吸引程度。我们采用如下两种计算策略：

- **Min**: 我们定义网页 w_i 对种子话题 C_s 的相似度为该网页与话题内所有网页相似度的最小值，如公式2.7。我们认为在一个种子话题的所有网页中，存在的那个与网页 w_i 最不相似的网页 w_j，如果连网页 w_j 与网页 w_i 的相似度都能取得较大值，那么整个种子话题 C_s 内的所有网页必然与网页 w_i 更加密切相关。即网页 w_i 就有更大的可能性归属于该种子话题 C_s。

$$S_{is} = \min(S_{ij}), \quad j \in C_s \tag{2.7}$$

- **AvgRate**: 我们定义网页 w_i 对种子话题 C_s 的相似度为该网页所带来的平均相似度对比种子话题 C_s 当前平均相似度的比例。如公式2.8。我们认为网页 w_i 如果能给种子话题 C_s 带来平均增加的相似度的比例越高，那么网页 w_i 与种子话题 C_s 必然更加密切相关。即网页 w_i 就有更大的可能性归属于该种子话题 C_s。

$$S_{is} = \frac{\text{Avg}(\sum_{j \in C_s} S_{ij})}{\text{Avg}(C_s)} \tag{2.8}$$

得到网页与种子话题的相似度后，我们将网页分配给相似度最高的 k 个种子话题。公式2.9实现了函数 $IsCut(\cdot)$，其中 th 使用了多层阈值来截断产生过完备的话题。阈值 th 在每一轮更新种子话题后重新赋值为当前种子平均相似度所处的层。比如说，之前种子话题的平均相似度 $Avg(C_s) = 0.76$，那么该种子话题所处的相似度层的阈值为 $th = 0.7$。如果新加的网页使得种子话题的平均相似度 $Avg(C_s \cup w_i) = 0.63$，那么由于更新后的种子话题不再属于原相似度层（即 $0.63 < 0.7$），启动阈值截断，即将之前的种子话题作为生成的完整话题加入到话题集合中去。同时，更新当前相似度层阈值为 $th = 0.6$。

$$IsCut(\cdot) = \begin{cases} 1, & \text{Avg}(C_s \cup w_i) < th \\ 0, & otherwise \end{cases} \tag{2.9}$$

基于上述两个算法，我们可以得到通过模拟 Lévy Walks 来生成话题的算法3。

算法 2 基于种子网页的网页多分配算法

Input: 相似度图 $G = (V, E, A)$，网页索引 $SortedIndex$，种子网页集合 SW，k

Output: 一系列话题集合 C

初始化空的话题集合 C
初始化种子话题 $C_s(s \in SW)$ 及对应的阈值 $th_s = 1$
for $i \in SortedIndex$ **do**
    计算网页 w_i 与每个种子话题 C_s 的相似度 S_{is}
    $\{i\} \leftarrow \underset{s \in SW}{\text{argmaxk}}(S_{is}, k)$ ▷ 取与网页 w_i 最相似的 k 个种子话题的索引
    for $s \in \{i\}$ **do**
        if $IsCut(C_s, w_i, th_s)$ **then**
            $C \leftarrow C \bigcup C_s$
            $C_s \leftarrow C_s \bigcup w_i$
            $th_s = \lfloor \text{Avg}(C_s) \times 10 \rfloor \div 10$
        end if
    end for
end for

算法 3 基于 Lévy Walks 的话题生成算法

Input: 相似度图 $G = (V, E, A)$，最相似网页个数 s，关联度集合 D, 分配话题数 k

Output: 过完备话题集合 C

for $d \in D$ **do**
    使用算法1　　**/* 寻找种子网页 */**
    使用算法2　　**/* 通过网页多分配算法生成话题 */**
end for

2.4.3 话题排序

一旦相似度图 $G(V,E,A)$ 构建完，我们通过算法3生成候选话题集合。然后我们在泊松噪声的假设下，使用泊松去卷积算法（Poisson Deconvolution，PD）来评估每个话题的权重：

$$\begin{aligned} w_{ij} &\sim \textbf{Poisson}(a_{ij}) \\ s.t. : w_{ij} &= \sum_{k=1}^{K} \mu_k C_{k_{ij}} \end{aligned} \tag{2.10}$$

其中 $C_{k_{ij}}$ 表示第 k 个话题是否同时包含网页 w_i 和 w_j。话题的兴趣度由 $i_k = \mu_k \cdot |C_k|$ 计算得到，其中 C_k 是第 k 个话题包含的网页数量。具体细节参见章节3。

2.4.4 时间复杂度分析

我们提出的算法3是基于 Lévy Walks 的话题生成算法。主要包含寻找种子网页和网页多分配算法两部分。采用多粒度种子策略，其中 D 是种子网页的关联度集合，集合 D 的个数通常小于 10。关联度越大，种子网页数量越少，通常从 $1 \sim 10$ 中选取。

在寻找种子话题过程中，我们需要在只保留 knn 个近邻的相似度图中计算每个网页的 SER 值以及过滤相关网页，时间复杂度分别为：

- 计算 SER：$O(s \cdot knn \cdot N)$；
- 过滤网页：$O(d \cdot knn \cdot N)$；

其中 N 是网页总数。knn 是每个网页要保留的近邻数，通常来说 knn 小于 100。s 是最相似的网页个数，通常小于 20；d 是要过滤的最相似网页个数（关联度），通常小于 10。而对于大规模网络数据而言，网页数量通常是巨大的。所以综上可以得到寻找种子网页的时间复杂度为近似线性的 $O(knn \cdot N)$。

在基于种子网页的网页多分算法中，我们需要针对每个网页计算两部分内容，分别是网页与种子话题的相似度以及网页分配给种子话题，时间复杂度分别是：

- 计算网页和种子话题的相似度：$O(|topic| \cdot |SW| \cdot N)$；
- 将网页分配给 k 个话题：$O(k \cdot N)$；

其中 N 是网页总数。$|topic|$ 表示话题内包含的网页个数，通常小于 100。$|SW|$ 是种子网页（种子话题）个数。k 是要分配的网页个数，通常小于 5。所以综上可以得到网页多分配算法的时间复杂度为 $O(|SW| \cdot N)$

从上面时间复杂度分析可以看出在算法3中，基于种子网页的网页多分配算法占据主要的时间复杂度。因此我们提出的 LWTG 算法的时间复杂度为 $O(|SW|\cdot N)$。种子网页的个数 SW 小于网页数 $\frac{N}{2}$。这个时间复杂度对于大规模的网络数据来说是非常高效的。

2.5 实验验证

本节对我们提出 LWTG 算法在 MCG-WEBV 和 YKS 这两个数据集上展开实验。主要进行两方面的比较。第一个是跟两个最好的能够处理噪声数据的聚类算法进行对比；第二个是跟其他三个最好的网络话题检测算法进行对比。通过这两类对比来验证我们算法的性能。

2.5.1 数据预处理

对于 MCG-WEBV 数据集，我们使用其中的文本数据包括标题、标签和描述。首先过滤掉文本数据中的停用词，由于该数据集基本由英文构成，所以使用 Python 中的 NLTK 模块，对每个单词提取词干、统计 tf-idf 值作为单词权重，最后由 tf-idf 值生成每个网页的特征向量。

对于 YKS 数据集，基本由中文组成。所以需要采用 NLPIR 系统对文本数据进行预处理，包括分词、去停用词、处理同义词和扩展词等，然后统计每个词的 tf-idf 的值，再对每个网页生成特征向量。

2.5.2 评测标准

在评测标准上，我们使用以下两种评测指标：

• 最高 10 个检测话题的 F_1 分数的均值-检测的话题数量（Top-10 F_1 v.s. Number of Detected Topics，NDT）：对于每个检测得到的话题 D_t，对应其最高匹配程度的真实标注的话题 G_t，我们可以定义话题精确度 $Precision$ 的公式2.11、话题召回率 $Recall$ 的公式2.12和话题 F_1 的分数公式2.13:

$$Precision = \frac{|D_t| \bigcap |G_t|}{|D_t|} \tag{2.11}$$

$$Recall = \frac{|D_t| \bigcap |G_t|}{|D_t| \bigcup |G_t|} \tag{2.12}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.13}$$

其中 $|\cdot|$ 表示一个话题中的网页数目。

• 准确率-平均到每个话题上的误检率（*Accuracy* v.s. False Positives Per Topics,*FPPT*）：准确率 *Accuracy* 的公式为2.14，*FPPT* 的公式为2.15:

$$Accuracy = \frac{\#Successful}{\#Groundtruth} \tag{2.14}$$

$$FPPT = \frac{\#Detected - \#Successful}{\#Successful} \tag{2.15}$$

其中话题被认为是正确检测到的标准由 NIR（Normalized Intersected Ratio）指标衡量。NIR 指标定义为公式2.16：

$$NIR = \frac{|D_t| \bigcap |G_t|}{|D_t| \bigcup |G_t|} \tag{2.16}$$

当检测到的话题的 NIR 高于一定阈值（通常设为 0.5）[xx] 时，我们认为这是一个正确检测到的话题。对于 #△ 表示对应集合 △ 的元素数量。

对于这两种评测指标，Top-10 F_1 v.s. NDT 衡量算法检测到的最好的前 10 个话题的性能，但是并没有考虑到检测过程带来的误检率。而 $Accuracyv.s.FPPT$ 综合衡量了所检测到的话题的准确率以及相应的平均每找到一个正确话题所带来的误检数。在这两种指标中，当具有相同 Top-10 F_1 分数或准确率时，更低 NDT 或 FPPT 的算法具有更优的话题检测性能。

2.5.3 实验设置

在实验中，我们选择了网页的文本数据进行词汇的 tf-idf 统计和编码，然后使用余弦距离构建相似度图。最后对每个网页只保留最相似的 *knn* 个网页的相似度值构建一个近邻图。这里对 MCG-WEBV 数据集的 *knn* 设为 100，最相似网页个数 s 设为 10。对 YKS 数据集，由于噪声相比 MCG-WEBV 数据集更严重，所以近邻值 *knn* 设为 15，最相似网页个数 s 设为 15。在所有实验中，同时在相似度图上过滤噪声网页的阈值 ϵ 设为 0.1。跟种子粒度相关的关联度参数集合 D 设为 $1, 2, 3, 4$。网页多分配的话题数 k 为 2。

2.5.4 与聚类算法的对比

我们对比了 LWTG 算法与两个性能最好的能够处理噪声数据的聚类算法：

a) Robust Spectral Clustering (RSC) for noisy data [xx]。这篇论文通过对相似度图的稀疏和隐式分解来处理噪声。然而，这种方法假设了噪声是稀疏的，但是在网络话题检测的场景下，大概 95% 的数据都是噪声数据。

b) Skinny-Dip (SD) [xx]。SD 基于检验分布是否为单峰分布的 Hartigan’s elegant dip test，从而得到一个有效的特征集来聚类。

注意到 RSC 和 SD 算法不是为了检测网络话题而是为了从噪声数据中进行聚类。所以对于 RSC，SD 和 LWTG 的对比，主要是为了验证能处理噪声数据的聚类算法，能否有效地在海量噪声数据中检测到网络话题。在下面的实验中，对 RSC，聚类个数设为真实话题个数。如 MCG-WEBV 数据集的 73 个真实话题和 YKS 数据集的 298 个真实话题。对于 SD，聚类个数由算法自动确定。

2.5.5 与网络话题检测算法的对比

我们对比了 LWTG 算法与 3 个性能最好的网络话题检测算法：

a) Multi-Modality Graph (MMG) [xx]。该算法属于多模态网络话题检测。Zhang 等人首先利用视频的 NDK 信息和文本信息建立相似度图 [xx]，然后使用图转移算法 [xx] 进行话题检测。MMG 假定了一个话题内的元素应该密切相关，所以，通常情况下 MMG 产生的话题规模较小。

b) PD with Non-negative Matrix Factorization on Graph (NMFG) [xx]。NMFG 和 LWTG 最大的不同的是产生话题的方式。NMFG 中使用基于图的非负矩阵分解（Non-negative Matrix factorization on Graph，NMG）[xx] 在相似度级联上生成过完备话题。NMG 在无噪声数据下的聚类非常耗时。

c) Latent Poisson Deconvolution (LPD) [xx]。LPD 算法在 MCG-WEBV 数据集和 YKS 数据集上均达到了当前最好的性能。相比单纯在 PD 上使用单一近邻图的方法，LPD 利用多个近邻图来排序话题。这个证明了我们的方法可以在不利用多个近邻图的情况下达到可接受的结果。

2.6 小结

第 3 章　网络话题的快速排序

3.1　引言

第 4 章　并行化处理

4.1　并行处理

第 5 章　总结与展望

5.1　本文工作总结

5.2　未来研究展望

附录 A　中国科学院大学学位论文撰写要求

学位论文是研究生科研工作成果的集中体现，是评判学位申请者学术水平、授予其学位的主要依据，是科研领域重要的文献资料。根据《科学技术报告、学位论文和学术论文的编写格式》（GB/T 7713-1987）、《学位论文编写规则》（GB/T 7713.1-2006）和《文后参考文献著录规则》（GB7714—87）等国家有关标准，结合中国科学院大学（以下简称“国科大”）的实际情况，特制订本规定。

A.1　论文无附录者无需附录部分

A.2　测试公式编号

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) = 0 \\ \frac{\partial (\rho \boldsymbol{V})}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}\boldsymbol{V}) = \nabla \cdot \boldsymbol{\sigma} \\ \frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\rho E \boldsymbol{V}) = \nabla \cdot (k \nabla T) + \nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{V}) \end{cases} \quad \ldots (A.1)$$

$$\frac{\partial}{\partial t} \int_{\Omega} u \, \mathrm{d}\Omega + \int_{S} \mathrm{n} \cdot (u \boldsymbol{V}) \, \mathrm{d}S = \dot{\phi} \quad \ldots (A.2)$$

A.3　测试生僻字

霜蟾盥薇曜灵霜🄵妙🄵虚🄵🄵澌菀枯菡萄🄵寥🄵冥🄵🄵🄵落🄵🄵便🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵飒🄵🄵俪🄵冤荸甲🄵藻卮言倥侗椒觞期颐夜阑彬蔚倥傯澄廓簪缨陟遐迤逦缥缃鹣鲽🄵懔闺闼璀错🄵婀嚕🄵🄵洞🄵🄵🄵缕🄵🄵逡巡🄵🄵🄵🄵🄵🄵踽踽🄵🄵氤氲瓠犀流眄蹀躞🄵🄵茕🄵璎珞螓首蘅皋🄵🄵缱绻昶皴皱颟顸愀然菡萄卑陬纯懿🄵🄵🄵🄵🄵🄵🄵🄵🄵墒墒墓🄵🄵🄵🄵🄵墚🄵坠🄵增🄵🄵🄵🄵🄵🄵🄵🄵墨墩🄵樽🄵🄵堕🄵🄵🄵🄵坟🄵🄵🄵🄵🄵🄵🄵墙擊🄵垦🄵🄵壁🄵🄵🄵壅🄵坛🄵🄵🄵🄵🄵🄵埙🄵🄵壑🄵压🄵壕🄵🄵垒圹垆🄵🄵🄵垄🄵🄵坊🄵壤🄵🄵🄵🄵坝🄵圭🄵🄵🄵🄵🄵娴🄵🄵🄵🄵妫🄵🄵🄵🄵🄵🄵🄵🄵娆嬉🄵娇🄵🄵🄵🄵🄵🄵🄵🄵🄵嫒嬗🄵嫱🄵🄵🄵🄵🄵🄵🄵嫒🄵🄵🄵嫔🄵🄵奶🄵🄵🄵婴🄵瓣🄵赢🄵🄵嬷婶🄵🄵🄵🄵🄵🄵🄵🄵孀🄵🄵娘🄵🄵🄵🄵🄵🄵🄵🄵娈🄵🄵🄵🄵🄵嫫婿媚🄵🄵嵯🄵🄵🄵🄵嵴🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵嵝嶂🄵崭🄵🄵岖🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵嶙🄵🄵🄵嶝🄵🄵峤🄵🄵🄵🄵🄵🄵峄🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵🄵岙🄵🄵嶷嵘🄵岭

参考文献

作者简历及攻读学位期间发表的学术论文与研究成果

作者简历:

姓名: 林尽忠　　性别: 男　　出生日期: 1992.2.27　　籍贯: 福建省古田县

2012.9-2016.6 在杭州电子科技大学通信工程学院获得学士学位

2016.9-2019.6 在中国科学院大学计算机科学与技术学院攻读硕士学位

已发表 (或正式接受) 的学术论文:

[1] Jinzhong Lin, Junbiao Pang, Li Su, Yugui Liu, Qingming Huang, ”Accelerating Topic Detection on Web for a Large-Scale Data Set via Stochastic Poisson Deconvolution”, in Proceedings of International Conference on Multimedia Modeling, 2019, pp. 590-602.

参加的研究项目:

[1] 2015 年 8 月 - 2017 年 4 月，面向网络事件的跨平台异质媒体语义协同与挖掘，国家自然科学基金重点项目。课题编号：61332016。

致　谢

时光荏苒，仿若白驹过隙。犹记得三年前独自一人来国科大面试，那情景，仿佛还发生在昨日，现在却到了要说再见的时候。这即将结束的三年北漂生活，同时也代表着学生生涯的结束。回望二十载的辛苦求学路，有近十载是独自异地求学，个中滋味，难以言表。在国科大的三年时光里，我不仅收获了很多知识，也得到了能力的提高和心理素质的锻炼。而这些，都离不开老师们的谆谆教诲，同学们的热情帮助以及家人朋友们的默默支持。在此，对你们致以衷心的感谢和祝福。

感谢我的父母。你们尽自己做大的努力给我提供良好的生活条件和求学环境，只愿我有更好的选择。从小到大，无论我做什么决定，你们总是无条件的支持我，鼓励我。相比学习成绩，更在乎我是否健康快乐。每每看到你们疲惫操劳的身影，我总是一阵心酸。只言片语无法表达我对你们的感谢和爱。祝愿二老身体健康，幸福快乐。

感谢黄庆明教授。在生活上对学生照顾有加，在科研上提供一流的设备，使我们能够心无旁骛地潜心科研。感谢您在面试阶段录取了我，给我打开一扇走入科研生活的大门，让我感受到学术的魅力。您严谨的科研态度、热情的关怀都使我铭记于心。在此也祝您身体健康，桃李满天下！感谢马丙鹏副教授，三年前，是您把懵懂的我招进中国科学院大学读研，从此走上科研之路，开始人生的新征程。感谢刘玉贵副教授在这三年对我的照顾，您严谨认真的治学态度和谦和豁达的人生态度着实令我钦佩。再次感谢马丙鹏老师和刘玉贵老师，祝您二位身体健康，事业顺利！

感谢庞俊彪副教授。感谢您这几年对我的指导和帮助。作为我的直接负责老师，您言传身教，真正做到了传道、授业、解惑。三年来，是您把我从一个科研门外汉，一步步带到门内。从课题的选择到算法研究，从实验开展到论文撰写，这其中的每一步都有您亲身参与，亲自指导。让我少走了很多弯路，同时也收获了很多。生活上，您对学生无微不至的体贴关怀；科研上，您对学生耐心有加，逐步指导。积极推动学生奋发向上，探索科研乐趣，不轻易放弃任何一个学生。您敏捷的思维逻辑、深刻的科研见解、深厚的学术功底、严谨的科研态度令我铭记于心。成为我不断学习，不断奋斗的目标榜样。感谢您带我度过这充实而难忘的三年时光。衷心祝愿庞老师阖家幸福，事业再上一层楼！

感谢实验室的许倩倩老师、王树徽老师、苏荔老师、李国荣老师、齐洪刚老师、李亮老师、张维刚老师和吴益灵师姐、杨智勇师兄、卓君宝师兄、吴哲师兄在学习中给我的帮助。感谢这三年唯一的室友和伙伴廖昌粟同学以及戚兆波、徐凯、辛永健、刘雪静、胡玲、郭双双等同学的陪伴，使得这三年的时光也有许多欢声笑语。在此祝愿所有的老师工作顺利，所有的同学们学业有成！

感谢未来的她，是你让我在迷茫困惑时有了坚持下来的动力！

最后，向百忙之中抽出宝贵时间评审本论文的专家和学者表示感谢！