流式光谱聚类

Shinjae Yoo
计算科学中心
布鲁克海文国家实验室
厄普顿，纽约州11973-5000
电子邮件：shinjae@gmail.com

郝黄
机器学习实验室
通用电气全球研究部
San Ramon，CA 94583
电子邮件：haohuanghw@gmail.com

Shiva Prasad Kasiviswanathan
三星研究美国
Mountain View，CA 94043
电子邮件：kasivisw@gmail.com

Abstract-Clustering是一种经典的数据挖掘任务发现数据中相似性的相互关联模式。 在许多现代领域，数据正在不断变化作为一条小溪。 出于可伸缩性的原因，将点聚类在数据流中需要设计单通道，有限的内存流聚类算法。 但是，表现了已知的流式聚类算法，通常使用K-在原始特征空间上的手段（或其变体）往往会受到影响当特征空间是高维的。 要克服这一点问题，我们提出了一种流光谱聚类算法。我们的算法保持归一化的近似值随着时间的推移拉普拉斯数据流并有效地更新在流媒体中改变这个拉普拉斯算子的特征向量时尚。 它只需要一次传递数据，消耗有限内存，并且对数据流的排序是稳定的。 我们为我们的流谱聚类提供理论分析算法和我们的实验结果表明，虽然获得了在可扩展性方面，其性能与其他已知产品相当批处理/流式聚类方法。

I.我 引言

聚类是一种重要的无监督学习技术通常它是现代使用的第一步数据分析。 任何良好的可扩展性的基本特征聚类算法是处理大量数据的能力高维特征空间中的数据。 最现代的高尺寸数据，如文档，图像和多媒体来自网络自然地以流媒体方式到达。 怎么样-永远，检测如此大量和高的集群尺寸数据流也是一个具有挑战性的问题原因如下：1）数据流可能是无限的，所以任何尝试存储整个的离线算法用于分析的流最终会耗尽内存，2）由于概念漂移，簇随时间动态演化，因此旧集群可能会合并，新集群可能会合并3）对许多流行的人来说，有一个维度的诅咒集群方法，4）各种在线应用，在（近）实时获得聚类很重要。

在本文中，我们研究了流式传输中的聚类设置。 虽然以前有相当多的文献流聚类算法[47]，[42]，[30]和大部分这些和其他流式聚类算法可以有效地完成处理大容量数据流，其性能趋于存在高维数据时降级[4]。 要克服这个问题，我们提出了光谱的流式自适应聚类算法。 光谱聚类获得了巨大的成功数据挖掘社区在过去十年中的受欢迎程度因为它能够发现嵌入式结构数据（又称歧管）。 在其最流行的形式，光谱聚类算法包括两个步骤：第一，特征向量使用核函数构造的归一化拉普拉斯算子用于嵌入数据集，第二，用于嵌入数据集聚类算法应用于嵌入数据集[40] [32]。

使频谱聚类适应的第一个挑战流设置在于规范化的构建拉普拉斯。 构造精确的归一化拉普拉斯算子是本质上是一个批处理操作，因为它需要访问整体数据和存储空间的长度为二次方流。 因此，传统的拉普拉斯建筑技术无法适应流媒体设置。 在本文中，我们专注于两个最流行的内核，余弦和高斯，对于这两个内核，我们提出了流式拉普拉斯算子近似技术。

下一个挑战在于更新频谱嵌入拉普拉斯的有效和高效，以便我们能够近乎实时地处理数据流。 为此，我们利用矩阵草图的最新想法 1保持低级别在每个时间步骤逼近整个观察数据，并且这种近似值作为新数据不断更新到达。 对于矩阵草图，我们采用最近的算法自由提出的（称为频繁方向）[27]。 运用这个低阶近似，在每个时间步，我们对齐学会了光谱嵌入之间具有相同的基础每两个相邻的时间步，这样就不会敏感概念漂移。 鉴于数据流的嵌入，我们然后应用流式K-means算法[41 处理创建和整合集群。

我们提出了我们的算法和理论分析表明在某些现实的假设下，我们构建了流嵌入是嵌入式的一个很好的近似 - 由昂贵的批处理技术创建的ding。 最好的我们的知识，我们的第一个流光谱聚类在真正的数据流设置中运行的算法（即，使用一次通过任何数据样本）。 我们建议的流媒体谱聚类算法是有效的以下方式：

（a）空间和时间有效，而只需要一个在内存占用有限的情况下传递数据。 让m是数据维度。 如果在时间t，n t点到达数据流，我们的算法只需要O（ml + mn t）空间和草图矩阵所在的O（max {mn t l，ml 2 }）时间尺寸m×l。 在实践中，设置l很多就足够了

1 非正式地，矩阵Z的草图是另一个较小的矩阵Z.大小比Z，但仍然接近它[27]。

与 n相比更… 因此，无论是时间还是空间要求几乎是输入大小的线性（作为输入在时间t是m×n t矩阵）。

（b）它很容易适应数据流上看不见的模式。 在每次执行步骤t，它都会提供更新的频谱嵌入

令S VD K（Z）= UΣkVk。 然后

秩（X）≤k 分 Z - X F = Z - U (k) Σ (k) V (k) F = Σ 米 J = K + 1 σ2 k + 1 。

到达时间的整个流数据集显示聚类列表
分配流中的所有数据点。

实证研究表明我们的方法是有效的与空间和时间相比，效率更高各种流行的批量/流式聚类算法来自文本，图像，网络和协作过滤的数据集域。

II。 B 背景

在本节中，我们将简要回顾一下基本概念在谱聚类和流K-means方法背后。我们从正式定义我们的符号开始。 我们表示[n] = 1：n。 向量总是以列为单位的方式用粗体字母表示。 对于向量v，v表示它转置和v表示其欧几里德范数。 对于矢量（α1，...，A M）∈R 米，DIAG（α1，...，A M）∈R 的m×m表示对角矩阵，其中 1，...， m为对角线条目。 我来吧表示维数为m×m的单位矩阵。 对于矩阵Z∈R M×N，行总和（Z）∈R m是具有1进入一个矢量等于Z的第i行中的条目总和。我们使用$z_{i,j}$表示Z的第（i，j）个元素。定义光谱范数如Z = sup {Zv：v = 1}。 我们也使用入门由Z_p表示的范数，其中p = 2给出（Frobenius范数）$Z_F^2 = \sum_{i,j} z_{i,j}^2$ 和p =∞给出 $Z_\infty = \max_{i,j} |z_{i,j}|$。给定一组矩阵，$Z_1$，...，$Z_t$，我们使用符号$Z_{[t]}$表示通过水平连接获得的矩阵$Z_1$，...，$Z_t$，即$Z_{[t]} = [Z_1，... Z_t]$。

我们使用S VD（Z）来表示奇异值分解 - Z的分数，即S VD（Z）=UΣV。 这里U是m×m正交矩阵，Σ是m×n对角矩阵，V是n×n正交矩阵。 Σ的对角线条目，其中σ1σ≥2≥...≥σ 米（给定米≤n）的，被称为Z的奇异值。我们按照惯例列出奇异值以非递增的顺序排列。 对于对称矩阵S∈Rm ×m，我们用E IG（S）来表示它的特征值分解，即UΛU= E IG（S）。 这里U是m×m正交矩阵和Λ是m×m对角矩阵（真实的）条目被λ1，...，λm的称为的本征值S（再次以非递增顺序列出）。

最佳秩k近似（在谱和谱中）Frobenius范数有义）到矩阵Z∈Rm ×n是$Z_{(k)} = \sum_{l=1}^{k} \sigma_l u_l v_l$，其中σl（l≤k）的前k个奇异Z的值，具有相关的左和右奇异向量U_l∈R m和V_l分别∈R N，。 我们使用S VD k（Z）来表示$Z_{(k)}$的截断奇异值分解，即，$Z_{(k)}$ = S VD k（Z）= $U_{(k)} Σ_{(k)} V_{(k)}$。 这里Σ DIAG（σ1，...，σk）的 ∈R K×K，$U_{(k)}$ 的 = [U 1，...，U K]∈R M×K和$V_{(k)}$ = [v 1，...，v k]∈Rn ×k 。 以下是众所周知的定理界定了最佳秩-k的近似误差近似。

定理2.1：[Golub et al。 [20]]设Z∈Rm ×n N> M，并且让σ1≥...≥σm为Z的奇异值

A.光谱聚类

算法1：S PECTRAL C LUSTERING [S PECTRAL C LUSTERING] 32]
输入：输入数据Y = [y_1，...，y_n]∈Rm ×n和
k∈R（簇数）
输出：n个实例的群集分配
构造核亲和度矩阵W∈Rn ×n，例如，
a）$w_{i,j} \leftarrow \exp\left(\frac{-\|y_i - y_j\|^2}{2\sigma^2}\right)$（高斯核），或
b）$w_{i,j} \leftarrow \frac{\langle y_i, y_j \rangle}{\|y_i\| \|y_j\|}$（余弦内核）
度矩阵D←diag（d（y_1），...，d（y_n））
其中点y_i，d（y_i）=Σ 的度数 $\sum_{j=1}^{n} w_{ij}$
归一化拉普拉斯L符号←I - D -1/2 WD -1/2
PΛP←ëIGk（L符号）
V←归一化P，单位L 2 行规范
将V行聚类成k个簇（使用K-means）

算法S PECTRAL C LUSTERING概述了一个非常受欢迎的基于嵌入数据集的聚类方法使用核函数，并利用顶部特征向量规范化拉普拉斯算子的发现，以发现潜在的群集TER值。 谱聚类与图形有很强的联系分区问题，因为本征空间用于解决放松形式的平衡图分区问题[32]。谱聚类的另一个方面是它可以捕获数据的多种结构，这是很难或不可能的通过K-means风格算法实现。 算法S PECTRAL - C LUSTERING首先建立亲和力或相似性矩阵（我们显示最流行的两个内核，余弦内核和高斯内核但它不仅限于这两个）。 然后我们构建一个拉普拉斯。 在算法S P 谱 C LUSTERING中，我们提出最流行的对称归一化拉普拉斯算子（L sym）。拉普拉斯语的其他流行选择包括非标准化Laplacian L un = D - W和随机游走归一化L rw = I - D -1 W.一旦我们计算拉普拉斯算子，我们就做了rank-k特征值分解得到谱嵌入限制在前k个特征值。 行标准化步骤在一个简单的聚类算法之前使用（例如当K-means）应用于V以识别聚类。

B.流式K-means

在本节中，我们将讨论最近的快速流式Euclidean Shindler等人的K均值聚类算法。 [41]（呈现在算法S TRM KM EANS中。 由于光谱聚类使用K-在我们提出的方法中，它意味着它的最终聚类步骤在构造的上应用这种流式K-means算法歧管。 如果输入数据点的数量是n和目标是输出k簇，算法维持一组微观表示为C的集群或设施，它们被合并到最后形成k簇。 算法S TRM KMS TEP，其中形成Algorithm S TRM KM EANS 的关键部分，详细说明如何将新数据点x添加到当前设施集C.概率δ/ f，其中δ是最小值的平方

x与C和C中任何设施之间的欧氏距离 f = 1 /（k（1 + log n））（记为设施成本），新设施初始化只有x。 剩余概率，x被分配给最近的现有设施。 算法试图确保不超过ρ=Θ（k log n）设备。 如果设施的数量达到ρ，那么算法重新组织（合并）设施以获得更小的设施数量并相应更新设施中心。 最后的“球K-means”步骤（这让人联想到执行类似Lloyd的重新聚类）以获得聚类中心。 Shindler等人。 [41 从理论上分析了算法S TRM KM EANS的 性能也提供了对其性能的实验支持。 但是，自从算法S TRM KM EANS在输入维度中运行，它会受到高维度的诅咒维度数据集。 但是，这不是我们的问题流光谱聚类方法（在下文中介绍）因为Streaming K-means算法在a上运行低维流形。

算法2：S TRM KM EANS（重述自[41]）
输入：数据流S，ρ是最大数量
设施，k是簇的数量，β是
标量。
输出：设施C和数据点分配
初始化f = 1 /（k（1 + logn））和空集C
而流S没完成呢

算法3：S TRM KMS TEP（重述自[41]）
输入：datapoint x，当前设施集C，ρ是
最大设施数，β是标量，f
是目前的设施成本
产出：新设施成本f，新设施集C和
数据点分配
测量δ= miny∈Cx - y 2（如果C为空则则δ= f）
设r是0到1之间的均匀随机数
如果r≤δ/ f那么
设置C←C∪{x}
其他
将x分配给C中最近的工厂
结束
而 C | >ρ
设f←βf
将每个z∈C移动到点的质心
分配给该群集
令w z为分配给z∈C的点数
C←C的第一个设施
对于每个z∈C做
设δ= miny∈Cz - y 2
如果发生概率δ/ f事件则
SetC←C∪{y}
其他
将z分配给其最近的C设施
结束

3 从流中读取下一个点x
4 [C，f]←S TRM KMS TEP（x，C，ρ，β，f）
5 结束
6 在加权点C上运行批次K-means
7 执行球K-means [9]在结果集群上
获得最终的聚类中心C.

III。 S TREAMING S PALAL C LUSTERING A LGORITHM

有两个主要困难需要克服
同时设计光谱聚类算法
流媒体环境：1）第一个挑战是如何实现
在流上方便地构造归一化拉普拉斯矩阵
（算法S PECTRAL C LUSTERING 的前三个步骤）。
拉普拉斯结构本质上是非流媒体任务
因为亲和度矩阵W需要访问整体
数据集，度矩阵D和拉普拉斯算子L.
因此，我们需要新颖的想法来近似拉普拉斯算子
流媒体设置中的矩阵。 2）给定拉普拉斯矩阵，
第二个挑战是构建流式流形
V，这很难，因为概念漂移（主题变化）
在流中导致这些嵌入变化很多
时间。 因此，我们需要用于鲁棒流式传输的新技术
嵌入式结构既有效又能够
适应固有的概念 - 漂移。

在介绍我们提出的算法之前，我们先来看看
描述我们的流设置。 我们假设数据到了
在流中，每个数据点都有一个指示的时间戳
当它到来时。 令S = {Y 吨 ∈R m×n个 T，T = 1，2，...}表示
一系列流数据矩阵，其中Y t代表
数据点在时间t到达。 这里m是特征的大小
空间，并且n 吨 ≥1是到达时间的数据点的数量

20 结束
21 更新C←C
22 结束

t。 2令Y [t] = [Y 1，...，Y t]∈Rm ×n [t]表示所有流
数据点到达时间t。

在本节的其余部分，我们将介绍和分析
我们的流式光谱聚类方法，在Algo-中概述
rithm S TRM SC。 我们首先描述各种建筑
我们算法的块。

A.归一化拉普拉斯结构

亲和矩阵。 亲和基质的尺寸增长
因为我们不断观察流中的数据，
我们不需要明确地构造亲和矩阵
用于谱聚类。 这里我们关注两个流行的内核
亲和力结构。

对于余弦内核的情况，在时间t，给定流
Y [t]与每列（点）具有单位L 2 -范数，亲和力
矩阵W cos = Y [t] Y [t] 。 令S VD（Y [T]）= UΣc ^ c V 中 ℃。我们可以
在不构造亲和力的情况下得到特征值分解
矩阵因为：

ÈIG（W COS）= E IG（Y [吨] Y [T]）= V c ^Σ2 c，V c。

换句话说，只要我们可以做流式奇异值
分解Y [t]，我们可以得到光谱嵌入V c
频谱聚类所需。

2 许多先前的流式算法假设只有一个点到达
每个时间步长，即，n T = 1。通过允许N t个为≥1，我们允许更
灵活的设置。

第4页

算法4：S TRM SC
输入：数据流S，l∈R，κ（≤l）∈R，ρ∈R，
k∈R，β∈R
输出：S中所有实例的群集分配
1 初始化f = 1 /（k（1 + log n））和空集C
2 s 0 ←全零矢量∈Rm
3 B 0 ←全零矩阵∈Rm ×l
4~ U 0 ←~U 1 （跳过第一次旋转）
5 当流S没完成的时候
6 令Y 吨 ∈R m×n个 吨是批次与时间戳t在
流S.
7 [B T，~U 吨（κ），V T，S T]←小号TRMëMB（Y T，B T-1，S T-1，κ）
8 V T，←归一化V T，与单元中的L 2行规范
9 R t ←~U t-1 ~U t (l)
10 用R t z 代替每个z∈C
11 对于每列x in V t do
12 [C，f]←S TRM KMS TEP（x，C，ρ，β，f）
13 结束
14 结束
15 在加权点C上运行批量K-means以形成k
集群

算法5：S TRM E MB
输入：Y 吨 ∈R m×n个 T，B T-1∈R M×L，S T-1∈R m和
κ∈R
输出：乙 吨，~U 吨（L），V T，S 吨
1 s t ←s t-1 +行和（Y t ）
2 c t ←s t / s t
3~ D t ←diag（<y 1，c t >，...，<y n t，c t >）
其中Y t = [y 1，...，y n t]
4~Y吨←ý吨~D -1/2
5 C t ←[B t-1，~Y t]
6~ U t（l）~Σt（l）~V t（l）←S VD l（C t）
7~Σ吨←诊断（√(~σ2 t 1 - ~σ2 t l)，...，√(~σ2 t l-1 - ~σ2 t l)，0）
其中~Σ 吨（1）= DIAG（~σT 1，...，~σT L）
8 B t ←~U t（l）~Σt（l）
9~ U t（κ）←[u 1，...，uκ]其中~U t（l）= [u 1，...，u l]
10~Σ吨（κ）←DIAG（~σT 1，...，~σ吨κ）
11 V吨←~Σ -1 t (κ) ~U t（κ）~ Y t

对于高斯核的情况，给定流Y [t]，
（i，j）亲和度矩阵的条目W gau等于exp（-y i -

来自[0，π]的均匀分布和余弦函数
入门应用。 根据[中的分析] 38]，作为
样本数d增加，这个随机傅立叶的误差
基数近似值为零。 以上投影即可
制定如下步骤：

1）从p（ω）中绘制di样本ω（1），...，ω（d）
ω~ 1 σ2 N（0，1）），和p（.）是快速傅立叶变换;
2）从均匀中绘制d 1样本（偏移）b 1，...，b d
在[0，π]上的分布;
3）计算投影数据，其中h（x）= cos（ωx+ b）;

由于每个点可以独立于其他点进行投影，
它非常适合流媒体应用。 这个想法是要取代
流中的每个点y的投影为h（y）。 设H [t] =
并[h（Y 1），...，H（Y N [吨]）]∈R d×N [t]的带S VD（H [T]）= UΣ 克 克 V G，

ÈIG（W GAU）≈ÈIG（H [T] H [T]）= V g Σ2 g V g 。

因此，高斯核可以被认为是应用余弦
H上的内核（忽略规范化）在下面，我们
假设每个流数据点都已使用转换
上述投影操作，因此，只能集中注意力
关于余弦核的情况。

度矩阵（算法S TRM E MB中的 步骤1-3 ）。 该
下一个问题是，当我们从流中观察到更多数据时，
对角度矩阵不仅在其大小上增加
但也在其条目的值（因为亲和力矩阵
W是非负的）。 我们使用聪明的人来克服这个问题
特技。 在时间t，流Y [t] 中的数据点y的程度
对于余弦内核，可以按如下方式计算：

$$d_t(y) = \sum_{z \in Y[t]} yz = y \sum_{z \in Y[t]} z = ys_t \quad (1)$$

其中s t = Σz∈Y [t] z =行和（Y [t] ）是数据集总和
矢量在时间t。 换句话说，没有构建亲和力
矩阵，只要我们知道，我们就可以计算度矩阵
整个数据集和向量。 在流设置中，给定s t，
我们可以计算出Y t中数据点的程度
流Y [t] = [Y 1，...，Y t]。 但是，正如我们观察到的那样
更多的数据点，s t 的范数不断增加，而在
时间，我们必须重新计算所有数据点的度数
在t之前到达。 为了克服这个问题，我们建议使用
数据集质心c t = s t / s t 对于程度近似为
如下。 限定，

$$\tilde{d}_t(y) = \frac{ys_t}{s_t} = yc_t\text{。} \quad (2)$$

请注意，在流式设置中，可以合理地假设
数据集质心c t = s t / s t （对于足够大的t）是

y i 2 (2σ2) 。 因此，W gau 的构造 要多一点
因为内核函数不是线性的，所以很复杂。 然而，
我们仍然可以通过首先使用线性化内核来进行近似
Rahimi和Recht的随机傅立叶基投影[38]。
这个想法是明确地将数据映射到欧几里德内部
使用随机特征图h的产品空间：$R^m \rightarrow R^d$
这样内核评估可以用
变换对之间的欧几里德内积。 CON-
cretely，对于任何两个数据点的x，y∈R 米，高斯
内核可以表示为h（x）h（y）的期望，
其中h（x）= cos（ωx+ b），ω∈Rm ×d来自a
适当地缩放高斯分布，B∈R d被拉

稳定（即，c t随时间变化缓慢），因为大多数话题
可以观察到分布。 以下引理
限制任何两个连续时间之间的质心转换
脚步。

引理3.1：令y∈R m为一个单位矢量。 转变了
以上定义的时间步长t之间的归一化y度
和t + 1满足：

$$\sim d_{t+1}(y) - \sim d_t(y) \leq n_{t+1} / (n_{[t+1]} - n_{t+1})$$

其中n_{t+1}是在时间t + 1到达的点数
$n_{[t+1]}$是直到时间t + 1观察到的总点数。

第5页

证明：

$$\begin{aligned}\backsim D_{T+1}(Y) - \backsim D_T(Y) &= YC_{T+1} - YC_{\text{吨}} \\ &\leq yc_{t+1} - c_t \\ &= \left\| \frac{s_{t+1}}{s_{t+1}} - \frac{\text{式}T}{\text{式}T} \right\| \\ &\leq \frac{s_{t+1} - s_t}{\text{式}T} \\ &= \frac{\sqrt{\ } n_{t+1}}{n_{[t+1]} - n_{t+1}}\end{aligned}$$

在这里我们使用y = 1.第二个不等式使用
事实上，s_t中的条目 ≥0并且以入口为主
由s_{t+1}。

上面的引理表明了一个点的归一化程度
一旦数据足够充足，流中的变化不大
已观察到，通常为$n_{[t]} \gg n_t$。 这表明
我们可以设置数据点y到达的归一化程度
在时间t as~ d_t（y）并且不在随后的时间调整它
脚步。 从上面的论证中，我们知道∽d_t（y）仍在继续
保持与实际标准化的良好近似
即使新数据到达，流中的y度也是如此。 我们用
这个归一化程度近似在其余部分
纸。

我们现在在实践中测试这种近似的质量。
图 1显示了两个不同的程度近似误差

0.25
0.2
0.15
0.1
0.05
0
0 5000 10000 15000
（a）按主题订购

0.25
0.2
0.15
0.1
0.05
0
0 5000 10000 15000
（b）按时间戳订购

图。1. 20上每个文档的度近似误差图
新闻组（20NG）数据集。 Y轴是两者之间的绝对误差之和
使用整个数据集近似度和真实归一化度。
图 1（a）显示了主题排序的流数据，这是最多的
难以接近的情况下，图 1（b）显示了订购的流数据
通过时间戳，这是更现实的情况。

20新闻组中的不同流订单（主题或时间戳）
（20NG）数据集（有关数据集的详细信息，请参阅
重刑 V ）。 batchsize（n_t）设置为1000.图1（a）显示

根据主题所有文档排序的流的结果
属于同一主题（群集）的（点）在一起。 这是
最难的情况是近似因为每当一个新的
主题介绍，它将创造一个重要的概念漂移。 在
尽管观察了流中明确的主题变化，程度
随着观察越来越多，近似误差减小
文档。 最终平均真实标准化程度为0.1388
但平均近似度数为0.1441，平均而言
度近似误差为0.0053（<4％）。 图 1（b）
显示更实际的案例，即订购文件的情况
通过他们真正的时间戳。 在这种情况下，程度近似
误差要小得多，平均为0.0009（<2％）。 有
在流的开头仍然有一些尖峰，但错误
随时间减少。

对于高斯核来说，通过使用来克服这些问题
我们可以使用前面描述的随机傅立叶投影
在投影空间上的相同程度近似。

归一化拉普拉斯算子（算法S TRM E MB中的 步骤4 ）。
构建亲和力矩阵和正常程度后 -
我们必须构造的特征向量
对称归一化拉普拉斯算子。 在这里，我们声称没有
明确地构建拉普拉斯算子，我们可以确定
L_{sym}的 特征向量 。 在批处理设置中，给定亲和力
矩阵W_{cos} = YY和相应的度矩阵D，
我们可以构造一个对称的归一化图拉普拉斯算子
如下：

$$L_{sym} = I - W_{sym} = I - D^{-1/2} W_{cos} D^{-1/2}$$

L_{sym}和。的特征向量之间存在等价性
我们在下面的引理中建立了W_{sym}。

引理3.2：L_{sym} 的第i个最小特征值是（1 -
λI），其中λ 我为W 符号的第i个最大特征值。

证明：从特征分解的定义，我们
我知道：

$$L_{sym} v = \lambda v。$$

我们可以将条款重新排列为以下形式：

$$(L_{sym} - \lambda I) v = 0$$

如果我们用L_{sym} 替换W_{sym}，那么我们有以下内容
这两者之间的关系：

$$\begin{aligned}(L_{sym} - \lambda I) v &= 0 \\ (I - W_{sym} - \lambda I) v &= 0 \\ (-W_{sym} + I - \lambda I) v &= 0 \\ (W_{sym} - I + \lambda I) v &= 0 \\ (W_{sym} - (1 - \lambda) I) v &= 0 \\ W_{sym} v &= (1 - \lambda) v\end{aligned}$$

由于W_{sym} 的特征值总是≥0并且因为
对于归一化，特征值从1到0，即
L_{sym}的 特征值范围从0到1 。 3

我们使用W_{sym} 的前k个有效特征向量
是L_{sym} 的k个最不重要的特征向量 。 仅限
因此，光谱聚类需要特征向量

3 在实践中，特征值的范围从-1到1.如果发生这种情况，我们可以
只需翻转特征向量和特征值的符号。

第6页

标准化拉普拉斯L sym的 谱嵌入，可以
很容易从YD -1/2 上的SVD获得，没有构造
拉普拉斯显式。

但是，在流设置中，构建精确
度矩阵是不可能的（没有存储所有观察到的
因此，我们使用流程度近似
等式的想法（ 2 ）。 设Y [t] = [Y 1，...，Y t]表示
原始输入流。 而不是Y [t]，我们专注于修改
流：

∽Y [T] = [∽Y 1，...，∽Y 吨]其中∽Y 我 = Y 我 ∽D -1/2 一世对于i∈[t]，

其中∽D i是数据点的度近似值
ÿ 我 。 注意that~Y [T]可以从Y [t]的以流来获得
时尚。

B.流媒体流形构造

流形构造 （算法中的步骤5-11）
S TRM E MB ）。 现在剩下的主要挑战在于
有效地构建流的频谱嵌入。
给定~Y [t]，呈现流式歧管结构
在算法S TRM E MB步骤5-11中。

我们使用基于矩阵草图的方法进行构建
以流式方式嵌入∽Y [t] 的频谱。 在他的重新
分纸[27]，Liberty引入了一种优雅的算法（称为
矩阵草图的频繁方向）。 频繁的指令 -
tions算法在流模型中运行并构造一个
草图矩阵使用（令人惊讶的）简单的"收缩"概念
一些正交向量。 在原始的频繁方向
算法设置[27]，输入是矩阵Z∈Rp ×d 。 在
每一步，Z行的一行由算法处理，并且
算法迭代地更新矩阵Q∈Rl ×d （l«p）
对于任何单元向量x∈R d，Zx时2 -qx 2≤2Z 2 F / l。
这里参数l调整了大小之间的权衡
草图矩阵和近似误差界限（较大的l
增加了计算和存储的要求
算法，同时给出更好的结果）。 最近，Ghashami和
飞利浦[18]，重新分析了频繁方向算法
表明它为低秩矩阵提供了相对误差界限
近似。

我们构建谱嵌入的方法
流~Y [t] （在Algo-步骤5-9之间概述）
rithm S TRM E MB ）基于扩展频繁指令
tions算法[27]在更广泛的环境中
每一步，我们添加n t »1个新列。 4和Frequent一样
方向，我们的算法只需要传递一次数据
流。 在算法S TRM E MB中，B t是维护的草图
stream Y [t] 的流程，并在每个时间步骤更新为新的
数据到了。 参数l（如前所述）定义
草图矩阵的大小和参数κ（≤l）定义
嵌入数据的维度。 我们讨论设置
之后这两个参数。

嵌入（重新）对齐（算法中的步骤9和10）
S TRM SC）。 虽然我们已经解决了这些挑战
构建embedding˘V 吨对于每个流批次中Algo-
rithm S TRM E MB，在时间t的低秩空间基础，∽U t （κ），趋向

4 基于类似草图的低秩矩阵近似方法是
最近在完全不同的特征选择环境中使用[23]。

由于概念漂移而随时间变化。 因此，我们不能
只需使用发现embedding˘V 吨集群。 每时每刻
步骤有必要重新调整过去的设施或微集群
质心到新的基础（算法S TRM SC中的步骤9-10）。
为此，我们构造对齐矩阵R t，如下所示：

R t = ∽U t-1 ~U t （κ） 。

然后我们先前通过R t z 旋转找到每个设施 z。

C.计算复杂性

在任何时间t，Algo-的运行时间
rithm S TRM E MB是O（max {mn t l，ml 2 }），使用电源 -
SVD的迭代或等级显示QR分解[20]。
空间复杂度为O（mn t + ml）。 对于批量光谱
聚类算法，在时间t使用所有数据直到时间
t，空间复杂度为O（mn [t] ），时间复杂度为
O（mn [t] k）。 一个人注意到算法S TRM SC很多
因为n [t]非常增长，所以比它的批处理对应物更有效
迅速。

D.理论分析

我们现在证明Algorithm S TRM E MB 的有效性，
通过在合理的假设下显示光谱
由算法S TRM E MB 构造的嵌入接近
利用整个数据流构建的嵌入。
由于空间限制，我们在此省略了详细的证明。

在算法S TRM E MB的 时间t，考虑秩-κ
近似C t = [B t-1，~Y t]。 让

小号 VDκ（C T） = ~U 吨 （κ） ~Σ 吨 （κ） ∽V T（κ）

算法S TRM E MB构造一个embedding˘V 吨 of~Y 吨 （原样
总结∽Σ -1 t （κ） 存在）定义为，

V T，= ∽Σ -1 t （κ） ~U t （κ） ~ Y t 。

现在B t-1是∽Y [t-1]的草图。 考虑秩-κ近似
~Y [t] = [~Y [t-1]，~Y t]。 让

小号 VDκ（∽Y [T]） = U [T]（κ）Σ[T]（κ） V [T]（κ）

我们比较算法所构建的embedding˘Vt
使用实际流~Y [t]构造的嵌入V t，
其中V t定义为（假设Σ -1 [t] （κ） 存在）

V t =Σ -1 [t] U [t] （κ）~Y t。

我们claim˘V 牛逼是第V的良好近似牛逼。

V 吨 - ˘V 五六 = V 吨 -˘V 吨 F
=Σ -1 [t] U [t] （κ）Y t - ∽Σ -1 t （κ） ∽U t ~Y（κ）F.
≤Σ -1 [t] U [t] （κ）- ~Σ -1 t （κ） ∽U t。F（κ） ~Y五六。 (3)

让我们专注于限制Σ -1 [t] U [t] （κ）- ~Σ -1 t （κ） ∽U tF。（κ）

Σ -1 [t] U [t] （κ）- ∽Σ -1 t （κ） ∽U tF（κ）
= U [t] （κ）Σ -1 [t] （κ）- ∽U t （κ） ∽Σ -1 t （κ）
= U [t] （κ）Σ -1 [t] （κ） - ∽U t （κ） Σ -1 [t] （κ） + ∽U t （κ） Σ -1 [t] （κ） - ∽U t （κ） ∽Σ -1 t （κ）
≤U [t] （κ）Σ -1 [t] （κ）- ∽U t （κ） Σ -1 [t] （κ） + ∽U t （κ）Σ -1 [t] （κ） - ∽U t （κ） ∽Σ -1 t （κ）
≤ù [T] （κ） - ~U吨Σ -1 [t] （κ） + √κ Σ -1 [t] （κ） - ~Σ -1 t （κ）
≤ù [T] （κ） - ~U吨Σ -1 [t] （κ） + κΣ -1 [t] （κ） - ~Σ -1 t （κ）∞。 (4)

第二个不等式使用Σ的事实 -1 [t] （κ） 和~Σ -1 t （κ） 是
对角矩阵，∽U t （κ） 的列是正交的。
对于最后的不等式，我们使用了自Σ以来的事实 -1 [t] （κ） - ∽Σ -1 t （κ）
是一个对角矩阵，它的Frobenius范数至多是正方形
其维度的根乘以最大对角线值。

因此，一个结合开V 吨 - ˘V 五六从各自边界如下
关于U [t] （κ） - ~U t （κ） Σ -1 [t] （κ） - ~Σ -1 t （κ）∞。对于前者，

命题3.4：

Σ -1 [t] （κ）- ∽Σ -1 t （κ）∞ ≤ ~Y [t] - ∽Y [t] F√（κ） / ˘σ 2 κ √(l - κ)
+最大 i∈[κ] | 1 / ~σ 吨 我 - 1 / √(~σ 2 吨 我 + t σ 2) |。 (6)

证明：请注意Σ -1 [t] （κ）- ~Σ -1 t （κ）∞ 可以绑定为

Σ -1 [t] （κ） - ~Σ -1 t （κ）∞ ≤ Σ -1 [t] - ~Σ -1 t （κ）∞ + Σ -1 [t] （κ） - ~Σ -1 t （κ）∞。

通过在Algorithm S TRM E MB中构建，

~Σ -1 t （κ） - ~Σ -1 t （κ）∞ =最大 i∈[κ] | 1 / ~σ 吨 我 - 1 / 吨 我 |
=最大 | 1 - 1 / √ |。

我们采用了最近对Huang等人的分析。[23]，谁分析
在使用矩阵草图的不同19行文中的类似边界
用于特征选择。

命题3.3（修改自[23]）：令λ 我表示
我 TH Y [t] ~ Y [t]的特征值。另外，假设L =
分钟 我= j的 |λ 我 - λ ĵ | > 0和满足

l =Ω (√m〜Y [t] - 〜Y [T] ‖F (κ) / L 2 ,

where〜Y [T] （κ）是秩κ近似of〜Y [T] 。然后

U t （κ） - 〜U t （κ） ≤ √2L / (L + 8〜Y 2 + √4L 2 + 16〜Y4[t]) 。 (5)

注意右边的（ 五 ）随着L的增加而减少。
在实践中，我们注意到这个很小。假设
在实践中通常满足L> 0，特别是如果m
比Y [t]中的数据点数量要小得多。
实际上，我们注意到设置l≈ √ 我就够了
取得好成绩。

绑定Σ -1 [t] （κ） - 〜Σ -1 t （κ），请注意

Σ -1 [t] （κ） - 〜Σ -1 t （κ） ≤ Σ -1 [t] （κ） - ˇΣ -1 t （κ） + ˇΣ -1 t （κ） - 〜Σ -1 t （κ） 。

让σ 吨 我，~σ 吨 我，和ˇσ 吨 我是第i个奇异值of~Y [T]，C 吨，和
分别为B t。这意味着

Σ -1 [t] （κ） = DIAG（1 /σ 吨 1，...，1 /σ 吨 κ），
〜Σ -1 t （κ） = DIAG（1 /~σ 吨 1，...，1 /~σ 吨 κ），和
ˇΣ -1 t （κ） = DIAG（1 /ˇσ 吨 1，...，1 /ˇσ 吨 κ）。

以下命题限制了Σ -1 [t] （κ） - 〜Σ -1 t （κ） 乘
边界Σ -1 [t] （κ） - ˇΣ -1 t （κ） 和ˇΣ -1 t （κ） - 〜Σ -1 t （κ） 分开。

i∈[κ] ~σ 吨 我~吨 2我 t + σ 2

绑定Σ -1 [t] （κ） - ˇΣ -1 t （κ），我们首先建立 〜Y [T] ÿ [T] ≽
B t B t跟踪[27]。这意味着对所有人而言
我∈[κ]，σ 吨 我 ≥ˇσ 吨 我。由Weyl的不等式[19]，对于∀i∈[κ]，σ 2
吨 我 ≤〜Y [T] ~Y [T] -ζ吨 2我 和 吨 2我 是本征值
分别为~Y [t] ~ Y [t]和B t B t）。现在

Σ -1 [t] （κ） - ˇΣ -1 t （κ） = 最大 i∈[κ] | 1/σ 吨 我 - 1/ˇσ 吨 我 |
≤ 最大 i∈[κ] |σ 吨 我 - ˇσ 吨 我 | / ˇσ 2 κ
≤ 〜Y [t] ~ Y [t] - B t B t / ˇσ 2 κ 。

通过使用[来自[分析] 23]，我们可以证明这一点

〜Y [t] ~ Y [t] - B t B t ≤ 〜Y [t] - 〜Y [t] （κ） ‖F √κ / l - κ 。

把所有东西放在一起，我们得到

Σ -1 [t] （κ） - 〜Σ -1 t （κ） ≤ 〜Y [t] - 〜Y [t] （κ） ‖F √κ / (ˇσ 2 κ l - κ)
+最大 i∈[κ] | 1/~σ 吨 我 - 1/~吨 2我 t + σ 2 | 。

请注意，对于右侧（ 6 ），这两个词都减少了
随着l的增加（对于第二项，更大的l导致更大
1 / ~吨 2我 t + σ 2

把它们放在一起。以下定理如下
命题 3.3和3.4与（ 3 ）（ 4 ）。

第8页

定理3.5：使用上面和下面定义的符号
命题 3.3的条件， 在每个时间t，嵌入 -
dingˇV 吨由算法S构造TRM ë MB满足，

V 吨 -V 五六 ≤〜Y 五六 (1/σ 吨 κ) √2L / (L + 8〜Y 2 + √4L 2 + 16〜Y4[t])
+ κ〜Y [t] - 〜Y [t] （κ） ‖F √ / (ˇσ 2 κ l - κ) + κmax i∈[κ] | 1/~σ 吨 我 - 1/~吨 2我 t + σ 2 | 。

上述定理表明，在合理的实践中
l，在每个时间t，嵌入的假设和设置
ˇV 吨由算法S构造TRM ë MB是靠近
嵌入V t使用整个数据流构建~Y [t]。

IV。 D 讨论

我们现在证明我们提出的Algo-的实用性是合理的
rithm S TRM SC简要地与一些相关的比较
方法。有关数据聚类的详细讨论
溪流，请参考[的调查] 7]，[17]，[37]，[2]。

全特征空间上的流聚类。基础的
流式聚类的想法是创建/更新微集群
同时为它们分配新的即将到来的数据流
最后使用摘要统计数据来生成最终的宏 -
集群。作业通常基于邻域
搜索完整的特征空间（例如在经典的K-中
装置/中位数）。Guha等人。[22]，[21 提出分歧 -
并且征服流聚类算法。奥卡拉汉等。
人。[36]提出了一种基于局部搜索的方法。Charikar等
人。[12]提出了K-median的随机算法
保证恒定因子近似的问题
一次通过和少量的目标函数
存储。提出了一种有效的流式文本聚类
由Zhong等人。[50]基于构建在线更新
用于球形K-平均值。

另一个课程是基于加速邻里
使用高效的数据结构进行搜索。张等人。[49]
构建了一个分层数据结构来逐步聚类
传入的数据流。Kranen等。[24 使用紧凑型

批量谱聚类比传统的K-更好
表示方法（及其变体）在完整功能上运行
空间。我们提出的第一个实验对比 V 。

预测空间上的流聚类。有相当的
一些关于在投影后聚类数据流的工作
低维空间。Aggarwal等人。[3]，[5]提出了一个
基于投影的聚类方法，保持统计
微集群数据流的信息，这些信息是
群集特征向量的极端扩展。这些想法是
进一步细化[4]。预计的聚类也是
与基于密度的算法一起[15]，[8]，[31]，[11]
[13]和高维数据[35]。的假设
预计的群集是让每个群集特定于
特征性特征子集，用于优化质量标准
簇。但是，功能子集可能完全不同
跨越不同的群集并且选择通常基于
参数设置，导致性能不稳定。
例如，[中的HDDStream算法] 35 试图建模
群集作为高密度区域（依赖于一些未知的
密度分布）。它总结了数据点和
这些点组合在一起的尺寸和
保持这些摘要随着时间的推移新点到达。
但是，它的性能依赖于很多参数
由用户指定。由于这些参数直接影响了
相关阈值，因此聚类结果是
对参数的选择敏感，相对排序
到达点和其他噪音问题。反之，
我们提出的方法是基于矩阵草图，其中
是整个的稳定压缩统计表示
（观察到的）数据流。因此，投影是稳健的
噪音和流的排序。我们提出了一个实验
节比较 V 。

进化/增量谱聚类。在一个不同的
不同的输入设置（来自我们的问题定义），进化 -
ary /增量谱聚类[14]，[33]，[34 接近
已经提出过，通常假设是这样的
图的全部或大部分点已经知道了
边缘是逐步添加的。虽然这些方法
在网络数据集上工作得很好，例如博客，这些方法
需要随时维护 2个图表。这很难
当数字时，用有限的内存保持这个矩阵
样本n很大，这通常是流式传输的情况

和自适应索引结构，用于维持流和新婚夫妇。阿克曼等人。[1]构造了一个小加权通过使用称为的新数据结构来获取数据流的样本核心树。Shindler等人。[41]最近提出了一个有效的流的高效算法（Algorithm S TRM KM EANS）使用在线设施位置文献中的想法进行聚类。他们的算法与现有算法相比都是有利的理论上和实验上。最近，Liberty等人。[28]提出了一种在线K-means聚类算法空间和时间要求只是多对数的流的长度。所有上述技术都在运行整个功能空间。但是，当数据集很高时维度，这些技术依赖于邻域搜索面部稳定性问题并且可能表现不佳，这是通常被称为维度的诅咒。Compa-我们的方法使用低维嵌入派生来自数据流的奇异值分解。那里-它具有类似的稳定性和性能优势

设置。据我们所知，还没有人知道这里的泛化是对包含的数据流进行操作的高维数据点。因此，我们不包括他们在我们的实验比较中。

高效的静态聚类。我们提出的算法主要 - 将一组集群中心候选者作为新的候选者更新并更新数据流来了。已经有许多有效的方法建议[39]，[46]，[44]，[6]，[48]，[29 试图恢复高维数据的稀疏聚类中心。但是全部它们需要访问整个数据集，因此不适合用于流式设置。因此，我们也不包括它们在我们的以下实验比较中。

V. E XPERIMENTAL A NALYSIS

在本节中，我们通过实验测试我们提出的ap-在效率和效率方面有所作为。所有的经验 -

第9页

表I. 小号 TATISTICS的实验数据集。

	数据集	#instances	#特征	#clusters
1	扇形	9619	55197	105
2	ohscal	11,162	11465	10
3	20ng的	18707	29476	20
4	RCV1	193844	47236	103
五	MNIST	70000	784	10
6	小	百万	3,072	75062
7	SenVec（SensIT车辆）	98528	100	3
8	滑稽演员	73421	100	86

在英特尔（R）Xeon（R）CPU X5650 2.67GHz上运行处理器，128GB内存。

A.实验设置

数据集。我们在四个文本数据集上进行了实验（扇区，ohscal，20Newsgroup和RCV1），两个图像数据集（MNIST和Tiny），一个传感器网络数据集（SenVec），一个协同过滤评级数据集（jester），所有摘要 - 表 I中列出了。对于前四个文本数据集，我们使用余弦核亲和矩阵，和高斯核亲和矩阵用于剩下的四个。四个文本数据集是L 2为每个文档标准化。

行业数据集由公司网页分类组成在工业部门的层次结构中。ohscal数据集是来自包含文件的OHSUMED集合来自医疗类别。20NG（20Newsgroup）是一个平衡的涵盖20个新闻主题的数据集。RCV1数据集包含制作手动分类的新闻专题报道的档案可由路透社[26]。我们使用了RCV1数据集的一个子集，只选择那些只有一个标签的文件。MNIST数据集有10类手写数字和784图像功能功能。所有上述数据集都可以在[10]或[16]。Tiny是非参数对象的大型Web图像集合和场景识别（从[下载] 45]）。从80年代开始百万个小数据集图像，我们创建了一个百万的数据集图像由60,000个标记图像组成，覆盖100个课程（来自[25]和其他随机抽样的图像。该此处的评估仅限于标记的60,000幅图像。我们使用了Tiny图像以及所有3072个原始特征，这是3色通道（RGB）中的32×32彩色图像，用图像来演示图像的聚类性能高维特征向量。SenVec（SensIT车辆）数据集包含车辆的分布式传感器网络数据分类。Jester数据集包含匿名评级数据用来推荐笑话。

基线。我们测试了我们提出的算法的两个版本：SSC-1，它使用Algorithm S TRM E MB来构造流的低维嵌入，但我们执行收集低维后的最终K均值聚类嵌入整个流; 和SSC-2，这是一样的算法S TRM SC（流K-means聚类是用于每一步）。SSC-1的结果证明了这一点光谱嵌入的质量，而SSC-2的结果表示完整的流谱聚类。

选择五种其他算法进行比较：经典算法K-means（简称KM），经典谱聚类算法 5（简称NJW），BIRCH [49]，流K-手段 6（简称SKM）和HDDStr [35]。首先两种算法是最广泛使用的批处理解决方案集群。BIRCH构建了一个分层的数据结构递增地传入数据流。讨论SKM早期是最近提出的快速准确的流媒体K-意味着聚类方法。HDDStr是一种有效的预测高维数据的流聚类方法。

评估指标和参数设置。我们用两个流行的评估指标：规范化的互信息（NMI）和纯度。为了显示每种算法的稳定性，我们还报告了这两个指标的标准差每个算法和每个数据集。

假设数据集中的簇k的数量为在其他作品中已知先验[32]，[41]，[35]。我们设定了N 吨为1000（N的稳定性试验吨将在第显示VD ）低谱嵌入的维数算法S TRM SC（κ）被设置为k的倍数。的大小矩阵草图l设置为 m如第 III-D部分所述 。对于使用高斯内核的随机特征映射，我们设置带宽比例为5000，投影尺寸（d）为2000.微集群（设施）的最大数量ρ在算法S中， TRM KM EANS设置为k log n，其中n为数据样本的数量（假设是先验已知的如[在[41]）。对于BIRCH，SKM和HDDStr，我们遵循建议在[49]，[41]，[35]各自的参数设置。对于所有比较算法，我们执行了WCSS K均值（最小化群内平方和，用30个内环和30个外环）以获得稳定的簇分配。

B.一般业绩比较

表 II显示了比较的一般性能算法。对于每个数据集，我们随机调整数据流顺序30次并报告平均NMI（上）和纯度（底部）。我们可以从中得出以下观察结果这些结果：

（1）与NJW相比，我们提出的SSC-1达到了92%NMI，平均纯度超过99%。这个结论我们的流光谱嵌入近似的公司接近计算上昂贵的质量批量版本。SSC-1具有非常相似的结果稳定性（使用标准偏差的测量）与NJW一样，平均而言，这比KM要好得多。

（2）SSC-1优于那些运行的算法高处的完整特征空间（KM，BIRCH和SKM）维数据集（四个文本数据集和Tiny图像数据集）。

（3）毫不奇怪，所有流式算法都能执行比批次K-means（KM）差。BIRCH，SKM和HDDStr平均只有30%的NMI和60%纯度与KM相比。但我们提出的SSC-2，其中在真正的流媒体设置中运行，平均达到58% NMI和97%纯度的KM，这是更有效的比其他流媒体竞争对手（在NMI中≈两倍）和1.6倍的纯度）。这证明了有效性SSC-2比其他流媒体竞争对手（BIRCH，SKM，和HDDStr）。

5 对于谱聚类，我们使用算法S PECTRAL C LUSTERING 。

6 对于流K-means，我们使用Algorithm S TRM KM EANS 。

第10页

（4）流媒体算法也更敏感
比批处理算法（NJW，KM等）不稳定
更改数据顺序。但是，通过利用低
维度嵌入，我们提出的SSC-2算法
实现与KM相当的稳定性。和....相比
无论是BIRCH，SKM还是HDDStr，SSC-2都是平均水平
在NMI中稳定三倍以上，大约十倍
纯度更稳定。

C.订单敏感度

众所周知，流式算法对此很敏感
数据的排序或概念漂移。测试性能
在这种情况下我们的方法，我们按数据流排序
以下两个附加条件，时间和主题顺序。
时间顺序反映了现实情景，而
主题顺序是模拟最困难的条件
由于大概念漂移而进行度近似
集群之间。对于后者，除了分组点
属于一个集群，我们也尽力保证
类似的主题彼此相邻（例如，在20NG数据集中，
comp.sys.ibm.pc.hardware主题中的文档集
和comp.sys.mac.hardware主题放在每个主题旁边
其他）。

图 2显示了按时间顺序排列的稳定性结果
输入流顺序。对于扇区数据集，只有少数几个集群
因此，在一开始就出现了较早的表现
比后者好。总的来说，所有三个数据集都显示出来
表现相当稳定。

图 3显示了按主题排序的输入流。外加
有条不紊地，通过分类的主题，SSC结果是均匀的
高于表 二的平均表现 。 我们推测
虽然有大量的主题变化，但数据流
草图可能更好地代表了因为
属于同一主题的文件在一起。 这个
导致更丰富的光谱嵌入，可以捕获更多
来自流的信息。一般来说，两个数字都表明了
对于我们提出的SSC-结果非常稳定
1和SSC-2算法，它们优于其他三种算法
流媒体算法，包括时间顺序和主题数据
排序。

D.批量大小灵敏度

不同批次尺寸（n t）的SSC-1和SSC-2测试是
如图 4所示 。 实验设置与那些相同
在表 II中，批量大小在500,800,1000,1200,1500，
1800和2000.性能表明稳定的行为
SSC-1和SSC-2具有不同的批量大小。

E.可扩展性比较

图 5显示了之间的可伸缩性比较
Tiny数据集上的各种聚类算法（
采样和下采样）。所有的流媒体算法
只需要几分钟即可集群超过一百万
数据点。特别是，我们观察到了我们的两个版本
SSC与其他高效流媒体相比毫不逊色
算法虽然具有更好的效果（表 二 ）。 COM的
与KM和NJW相比，我们提出的算法表明
卓越的可扩展性，同时具有可比性。

10 6
10 4
10 2
时间 (秒)
10 0
10 -2
10 4 10 5 10 6 10 7
数据集大小（n）
SSC-2
SSC-1
SKM
桦木
HDDStr
KM
NJW

图5.可扩展性实验。KM和NJW不会扩展到大型数据集（由于其极高的内存开销而失败）。

VI。 C 结论

我们提出了一种流式光谱的新方法
集群。我们的算法构建了一个简洁的近似值
拉普拉斯算子，并从矩阵草图到适应思想
有效地构建了一个低维谱嵌入
流。我们的算法只需要一次通过
数据，利用有限的存储，很好地适应概念漂移
并近乎实时地运作。实验结果表明
我们提出的方法可以胜过其有效性
高维流行的流媒体聚类算法
数据流同时实现良好的可扩展性。

R EFERENCES

[1] M. R Ackermann，M。Märtens，C。Raupach，K。Swierkot，C。Lam-梅森和C.索勒。Streamkm ++：数据的聚类算法流。ACM JEA，2012。
[2] CC Aggarwal。流聚类算法调查，2013。
[3] CC Aggarwal，J。Han，J。Wang和PS Yu。一个框架聚集不断发展的数据流。在VLDB，2003年。
[4] CC Aggarwal，J。Han，J。Wang和PS Yu。一个框架投影的高维数据流聚类。在VLDB，2004年。
[5] CC Aggarwal和PS Yu。用于群集大量文本的框架和分类数据流。在SIAM SDM，2006年。
[6] M. Azizyan，A。Singh和L. Wasserman。Minimax理论的高 - 具有稀疏平均分离的尺寸高斯混合物。在NIPS中，2013。
[7] D.巴巴拉。集群数据流的要求。ACM SIGKDD探索通讯，2002年。
[8] C. Bohm，K。栏杆，HP Kriegel和P. Kroger。密度连接使用本地子空间首选项进行聚类。在IEEE ICDM，2004年。
[9] Vladimir Braverman，Adam Meyerson，Rafail Ostrovsky，Alan Royt-男子，Michael Shindler和Brian Tagiku。良好的流式k-means可群集数据。在SODA，第26-40页，2011年。
[10] D.蔡。用于特征学习的Matlab代码和数据集。 HTTP：//www.cad。zju.edu.cn/home/dengcai/Data/data.html 。
[11] F. Cao，M。Ester，W。Qian和A. Zhou。基于密度的聚类带有噪音的不断发展的数据流。在SIAM SDM，2006年。
[12] M. Charikar，L。O'Callaghan和R. Panigrahy。更好的流媒体聚类问题的算法。在ACM STOC，2003年。
[13] Y. Chen和L. Tu。用于实时流数据的基于密度的聚类。在ACM SIGKDD，2007年。
[14] Y. Chi，X。Song，D。Zhou，K。Hino和BL Tseng。发展的通过结合时间平滑度进行谱聚类。在ACM SIGKDD，2007。
[15] M. Ester，HP Kriegel，J。Sander和X. Xu。基于密度的算法用于在具有噪声的大空间数据库中发现聚类。在ACM SIGKDD，1996。
[16] RE Fan和CJ Lin。Libsvm数据：分类，回归和多标签。 http://www.csie.ntu.edu.tw/〜cjlin / libsvmtools /数据集/ 。

第11页

表二。 P ERFORMANCE比较 NMI（TOP）及 P URITY（BOTTOM）。 Ť HE NUMBERS中括号里的标准偏差分类指标。w ^ èCOULDN '牛逼运行NJW ON THE RCV1及牛逼 INY数据集DUE TO OUT MEMORY运行。ť HE号码NJW ON THE RCV1 DATASET从[43]。

	NJW	SSC-1	KM	桦木	SKM	HDDStr	SSC-2
扇形	0.4039 (0.008)	0.3961 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.008)
	0.9794 (0.001)	0.9790 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.002)
ohscal	0.3408 (0.018)	0.3217 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.018)
	0.8897 (0.004)	0.8833 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.009)
20ng的	0.4836 (0.008)	0.4699 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.010)
	0.9643 (0.002)	0.9532 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.002)
RCV1	[43] 0.2875 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.005)
	- (-)	0.9432 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.003)
MNIST	0.4965 (0.010)	0.3792 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.003)
	0.8868 (0.003)	0.8788 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.002)
滑稽演员	0.5099 (0.003)	0.4723 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.012)
	0.9444 (0.001)	0.9442 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.007)
SenVec	0.3007 (0.001)	0.2810 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.021)
	0.6769 (0.001)	0.6921 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.062)
小	- (-)	0.2001 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.001)
	- (-)	0.9569 ([illegible])	[illegible]	[illegible]	[illegible]	[illegible]	[illegible] (0.001)

0.6 0.5 0.4 0.3 0.2 0.1 0 NMI
SSC-1 SSC-2 SKM 松木 HDDStr
0.4 0.35 0.3 0.25 0.2 0.15 0.1 0.05 0 NMI
0.6 0.5 0.4 0.3 0.2 0.1 0 NMI
2000 4000 6000 8000 10000 1000 2000 4000 6000 8000 10000 1000 3000 5000 7000 9000 11000 13000 15000 17000 19000
按时间顺序记录索引 按时间顺序记录索引 按时间顺序记录索引
（a）部门 （b）ohscal （c）20NG

图2.概念随时间的漂移。我们的SSC-1和SSC-2都胜过其他三种流媒体方法。

0.7 0.6 0.5 0.4 0.3 0.2 0.1 0 NMI
SSC-1 SSC-2 SKM 松木 HDDStr
0.4 0.35 0.3 0.25 0.2 0.15 0.1 0.05 0 NMI
0.7 0.6 0.5 0.4 0.3 0.2 0.1 0 NMI
2000 4000 6000 8000 10000 1000 2000 4000 6000 8000 10000 1000 3000 5000 7000 9000 11000 13000 15000 17000 19000
主题顺序的文档索引 主题顺序的文档索引 主题顺序的文档索引
（a）部门 （b）ohscal （c）20NG

图3.跨排序主题的概念漂移。我们的SSC-1和SSC-2都胜过其他三种流媒体方法。

[17] MM Gaber，A。Zaslavsky和S. Krishnaswamy。挖掘数据流：回顾。ACM Sigmod Record，2005。

[18] M. Ghashami和JM Phillips。确定性低的相对误差秩矩阵近似。在SODA，第707-717页，2014年。

[19] GH Golub，M。Heath和G. Wahba。广义交叉验证为选择良好脊线参数的方法。技术计量学，21（2）：215-223,1979。

[20] GH Golub和C. F Van Loan。矩阵计算，第3卷.JHU按，2012。

[21] S. Guha，A。Meyerson，N。Mishra，R。Motwani和L. O'Callaghan。聚类数据流：理论与实践。IEEE TKDE，2003。

[22] S. Guha，N。Mishra，R。Motwani和L. O'Callaghan。聚类数据流。在计算机科学基础。IEEE，2000。

[23] H. Huang，S。Yoo和S. Kasiviswanathan。无监督功能选择数据流。在CIKM，2015年。

[24] P. Kranen，I。Assent，C。Baldauf和T. Seidl。clustree：索引用于任何时间流挖掘的微集群。知识和信息系统，2011。

[25] A. Krizhevsky，V。Nair和G. Hinton。cifar-100数据集。HTTP：//www.cs.toronto.edu/〜KRIZ / cifar.html 。

[26] DD Lewis，Y。Yang，TG Rose和F. Li。Rcv1：一个新的基准文本分类研究的集合。ACM SIGKDD探索时事通讯，5：361-397，2004年。

[27] E. Liberty。简单而确定的矩阵草图。在ACM SIGKDD，第581-588页，2013年。

[28] E. Liberty，R。Sriharsha和M. Sviridenko。一种在线算法k均值聚类。arXiv preprint，2014。

[29] J. Liu，C。Wang，M。Danilevsky和J. Han。大规模光谱在图上聚类。在第二十三届国际会议论文集中人工智能联合会议，第1486-1492页。AAAI按，2013年。

[30] C. Maung和H. Schweitzer。通过效率无监督功能选择。在NIPS，2013年。

[31] O. Nasraoui和C. Rojas。强大的聚类，用于跟踪噪声演变数据流。在SDM，2006年。

[32] A. Ng，M。Jordan和Y. Weiss。关于谱聚类：分析与分析算法。在神经信息处理系统的进展14，第849-856页，2002年。

[33] H. Ning，W。Xu，Y。Chi，Y。Gong和TS Huang。增量光谱集群应用于监控不断发展的博客社区。在SIAM SDM，2007年。

[34] H. Ning，W。Xu，Y。Chi，Y。Gong和TS Huang。增量光谱通过有效地更新特征系统来聚类。模式识别，43（1）：113-127,2010。

[35] I. Ntoutsi，A。Zimek，T。Palpanas，P。Kröger和HP Kriegel。密度-

第12页

0.4 0.38 0.36 0.34 0.32 0.3 0.28 0.26 0.24 0.22 NMI
SSC-1 SSC-2
0.34 0.32 0.3 0.28 0.26 0.24 0.22 0.2 NMI
0.48 0.46 0.44 0.42 0.4 0.38 0.36 0.34 0.32 NMI
500 800 1000 1200 1500 1800 2000 500 800 1000 1200 1500 1800 2000 500 800 1000 1200 1500 1800 2000
批量大小（nt） 批量大小（nt） 批量大小（nt）
（a）部门 （b）ohscal （c）20NG

0.28 0.26 0.24 0.22 0.2 0.18 0.16 NMI
SSC-1 SSC-2
0.4 0.35 0.3 0.25 0.2 0.15 NMI
0.5 0.45 0.4 0.35 0.3 0.25 0.2 NMI
500 800 1000 1200 1500 1800 2000 500 800 1000 1200 1500 1800 2000 500 800 1000 1200 1500 1800 2000
批量大小（nt） 批量大小（nt） 批量大小（nt）
（d）RCV1 （e）MNIST （f）小丑

0.3 0.25 0.2 0.15 0.1 NMI
SSC-1 SSC-2
0.21 0.2 0.19 0.18 0.17 0.16 0.15 NMI
500 800 1000 1200 1500 1800 2000 500 800 1000 1200 1500 1800 2000
批量大小（nt） 批量大小（nt）
（g）SenVec （h）微小的

图4.不同批量大小的方法的稳定性。SSC-1和SSC-2都可以通过改变批量大小来表现出稳定的性能。

基于高维数据流的预测聚类。在SIAM SDM，2012。

[36] L. O'callaghan，A。Meyerson，R。Motwani，N。Mishra和S. Guha。用于高质量聚类的流数据算法。在IEEE ICDE中，2002年。

[37] ME Orlowska，X。Sun和X. Li。可以独占聚类流数据实现？ACM SIGKDD探索通讯，

[49] T. Zhang，R。Ramakrishnan和M. Livny。Birch：一个有效的数据非常大型数据库的聚类方法。在ACM SIGMOD，1996年。

[50] S. Zhong。高效的流式文本聚类。神经网络，18（5）：790-798,2005。

2006年。
[38] A. Rahimi和B. Recht。大规模内核的随机特性
机器。NIPS，2007年。
[39] D. Sculley。网络规模的k均值聚类。在ACM WWW，2010年。
[40] J. Shi和J. Malik。标准化剪切和图像分割。 IEEE
TPAMI，22（8）：888-905,2000。
[41] M. Shindler，A。Wong和AW Meyerson。快速准确的k-means
对于大型数据集。在NIPS，2011年。
[42] Q. Song，J。Ni和G. Wang。基于快速聚类的特征子集
高维数据的选择算法。IEEE TKDE，2013年。
[43] Y. Song，W。Chen，H。Bai，C。Jin和EY Chang。平行光谱
集群。数据库中的机器学习和知识发现，
2008年。
[44] W. Sun，J。Wang，Y。Fang，et al。正规化的k均值聚类
维数据及其渐近一致性。电子期刊
统计，2012年。
[45] A. Torralba，R。Fergus和WT Freeman。8000万张小图片：
用于非参数对象和场景识别的大型数据集。 IEEE
TPAMI，2008年。
[46] J. Wang，J。Wang，Q。Ke，G。Zeng和S. Li。快速近似k均值
通过群集闭包。在IEEE CVPR，2012年。
[47] H. Yang，MR Lyu和I. King。多任务的高效在线学习
功能选择。ACM TKDD，2013年。
[48] J. Yi，L。Zhang，J。Wang，R。Jin和A. Jain。单通道算法
有效地恢复高维数据的稀疏聚类中心。
在ICML，2014年。