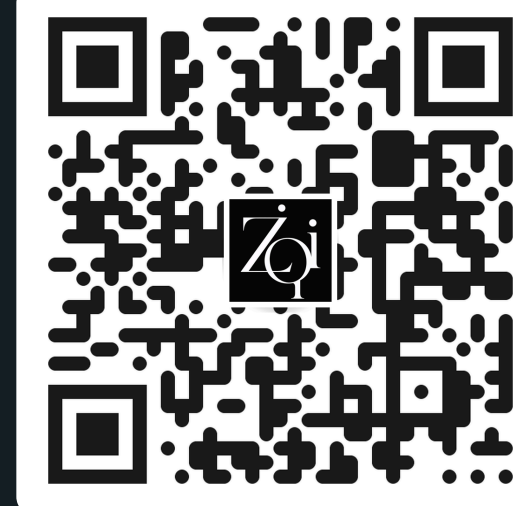
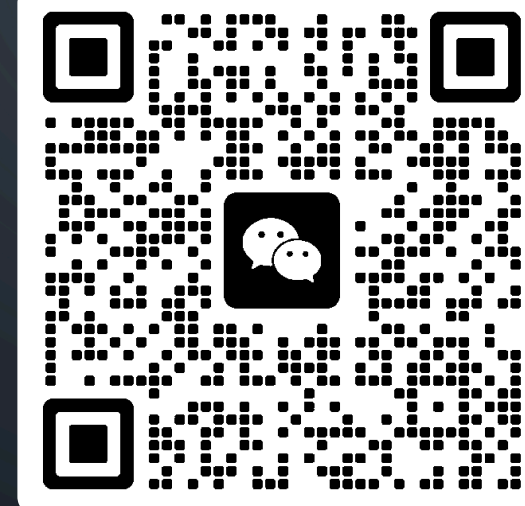


面向类别过滤近似最近邻搜索的高效索引

Efficient Index for Category Filtering Approximate Nearest Neighbor Search

王紫萁 张景哲 胡伟*

计算机软件新技术全国重点实验室（南京大学），江苏 南京 210023 · *whu@nju.edu.cn

NANJING
UNIVERSITY论文全文
PDF 下载个人主页
ziquiwww.github.io微信交流
扫码添加

❶ 研究背景与问题

- 向量近似最近邻搜索（ANNS）已是 RAG 等智能检索的基础功能，但单一向量搜索难以满足现实查询需求。
- 典型需求是向量+结构化数据的过滤搜索：每个向量有一个类别属性，类别构成树型或有向无环图型包含关系。
- 该问题称为类别过滤近似最近邻搜索（CFANNS）：给定查询向量 q 与查询类别 p ，返回最相似的 k 个向量且类别须为 p 或其子类，典型场景如 RAG 中只在目标文件夹下抽取文本、密级语料中只抽取有权访问的文件。
- 通用过滤索引 ACORN 与标签过滤索引 UNG 未针对类别层次优化，无法取得最优查询性能。

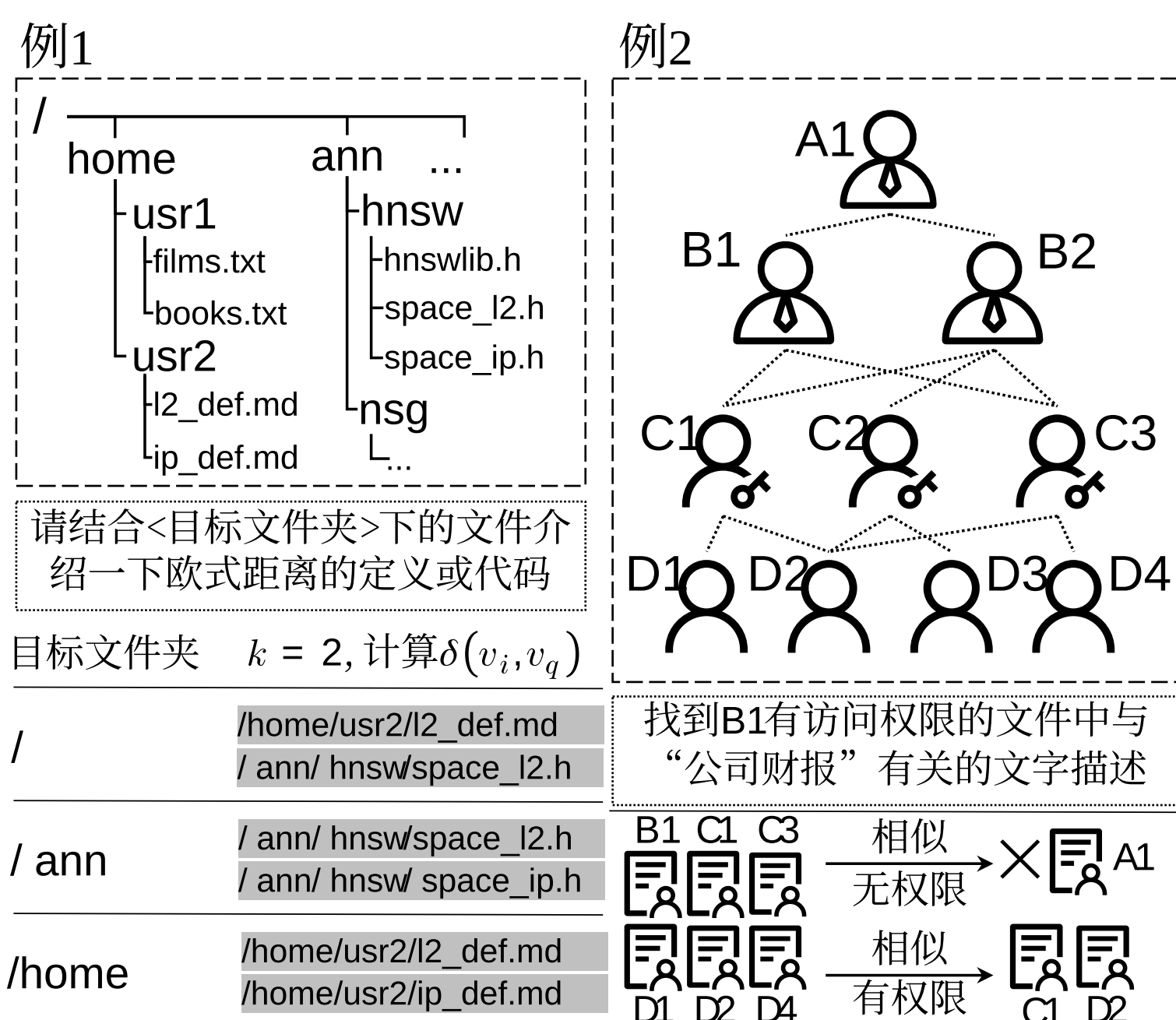


图 1 CFANNS 应用举例：文件夹归属（例 1）与文件密级（例 2）。

❷ 问题定义

定义 1 近似最近邻搜索（ANNS）

给定距离度量 $\delta(\cdot)$ 、向量数据集 V 、查询 q 与近邻数 k ，抽取 $|S|=k$ 的子集以最小化 $\sum_{v \in S} \delta(v, q)$ ；准确率用召回率 $\text{recall} = |S \cap R| / |R|$ 衡量。

定义 2 类别过滤近似最近邻搜索（CFANNS）

混合数据集 $D = \{V, C\}$ ，映射 $c(\cdot): V \rightarrow C$ 给出向量的唯一直接类别；类别拓朴为有向树 T 或有向无环图 A 。给定 q 与 $p \in C$ ，抽取 $|S|=k$ 的子集以最小化 $\sum \delta(v, q)$ ， $c(v) \leq p$ 。

❸ 核心思路：从 $O(|C|)$ 到 $O(\log |C|)$

朴素思路

为每个类别构造一张先知近邻图（仅在该类别及其子类包含的向量上建图）。需要 $O(|C|)$ 张图，时间与空间开销巨大，无法应用于百万向量与大量类别场景。

关键观察

不同类别的先知近邻图之间存在大量重复的边——两个类别的图中可能有大量点具有完全相同的邻居集合。

本文思路：将 $O(|C|)$ 个先知近邻图无损压缩到 $O(\log |C|)$ 的单一索引——类别增强近邻图（CatGraph）：把各图压到同一张更稠密的图中、每条边仅出现一次，查询时动态选择邻居列表拟合先知近邻图上的搜索。

❹ 方法：类别增强近邻图

核心思想：每个向量沿类别树的路径维护多个压缩邻居列表，从而在一张图上同时表达所有祖先类别对应的先知近邻图。

- 向量 v 的压缩邻居列表个数 = 类别树根 C_R 到 $c(v)$ 的路径长度；列表 N_w^C 仅包含子图 G_C 中的点，天然满足类别要求。
- 查询任意类别 p 时，只需读取该类别对应的压缩邻居列表，即可复现 G_p 上的搜索。

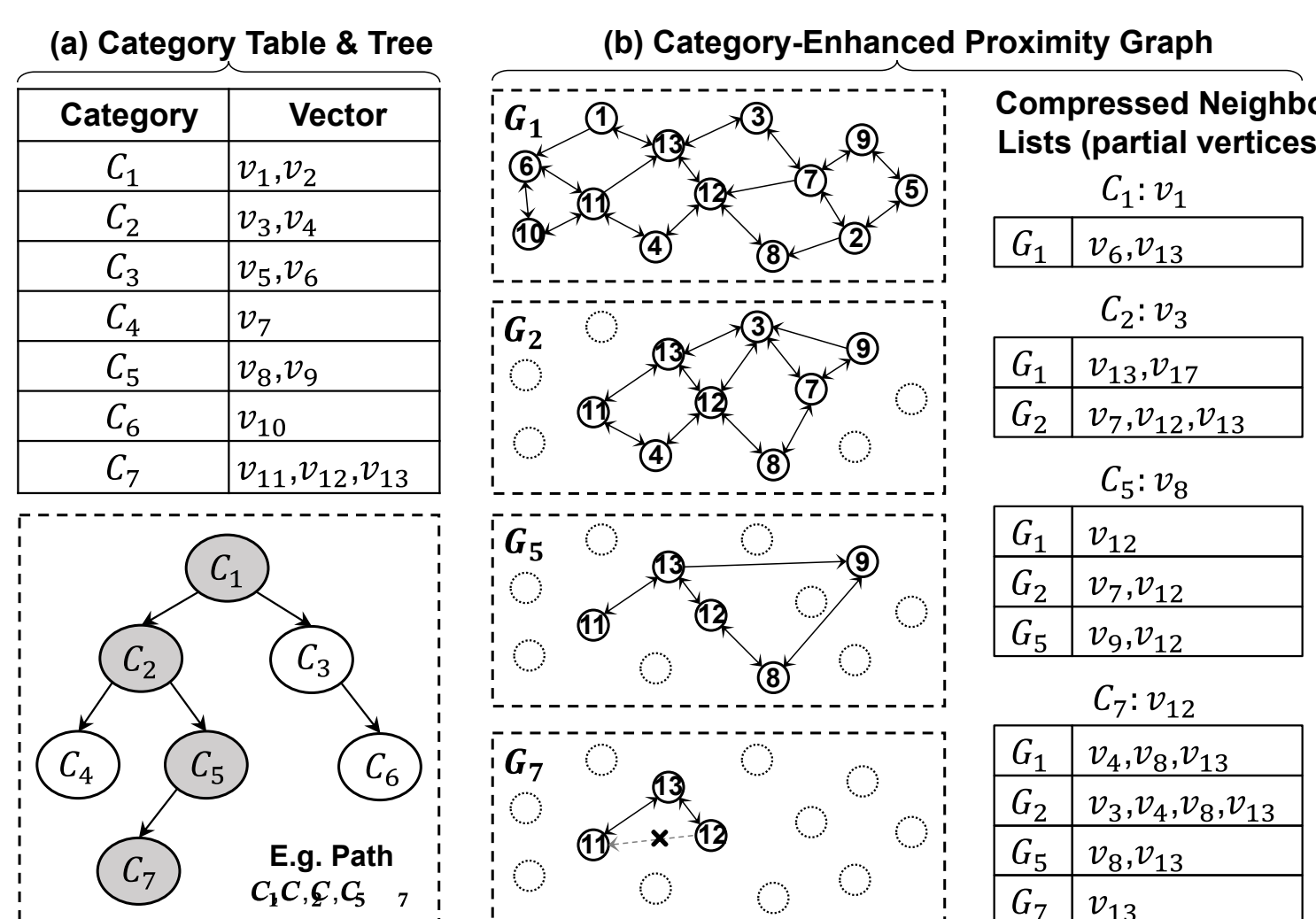


图 2 索引结构概览：(a) 类别映射与类别树；(b) 类别增强近邻图与部分向量的压缩邻居列表。

算法 1. 类别增强近邻图的向量插入算法 (Insert)

输入：待插入向量 v ；向量所属类别 C_t

输出：新向量插入后的索引 G

索引参数：构建时集束搜索宽度 ω_c ；近邻图最大出度 m ；类别映射 $c(\cdot)$ ；类别关系 $T: \leq$

- $U \leftarrow \{\}$; // 候选邻居集合
- $C_R \leftarrow$ root category of T ; // 类别树中的根类别
- foreach category C on path $C_R \rightarrow C_t$ in T do
- $U \leftarrow \{v \mid v \in U \wedge c(v) \leq C\}$;
- $ep \leftarrow$ a random vector where $c(ep) \leq C$;
- if $|U| < m$ then $U \leftarrow U \cup \text{CatCA}(v, C, \omega_c, ep)$;
- $N_v^{G_c} \leftarrow \text{Prune}(V, U, \frac{m}{2})$;
- foreach neighbor $u \in N_v^{G_c}$ do
- $N_u^{G_c} \leftarrow N_u^{G_c} \cup \{v\}$;
- if $|N_u^{G_c}| > m$ then $N_u^{G_c} \leftarrow \text{Prune}(u, N_u^{G_c}, m)$;
- $c(v) \leftarrow C_t$;
- return G ;

算法 1 类别增强近邻图的向量插入算法（Insert）。

理论保证

- 定理 1 忠实性** CatCA 运行任意时刻，候选集中的向量类别恒被目标类别覆盖。
- 定理 2 无损压缩** 对所有类别的先知近邻图无损压缩，任意查询都能达到基于图索引的性能上界。
- 定理 3 空间** $O(m n \bar{h})$ ； T 相对平衡时 $\bar{h} = \log |C|$ 。
- 定理 4 构建时** 构建最坏时间复杂度 $O(m d \bar{h} n \log n)$ 。
- 定理 5 搜索时** 搜索最坏时间复杂度 $O(m d \log n')$ ， n' 为满足类别过滤的向量子集大小。

算法 2. 类别感知候选邻居选取过程 (CatCA)

输入：目标向量 v ；目标类别 C_t ；集束搜索宽度 ω ；入口点 ep

输出：候选邻居集合 U

- ```
// 集束窗口小顶堆 W ，候选邻居大顶堆 U
1 $W \leftarrow \{ep\}, U \leftarrow \{ep\}$;
2 while $|W| \neq \emptyset$ do
3 $w \leftarrow$ pop one vector from W ;
4 foreach unvisited neighbor $u \in N_w^{G_c}$ do
5 mark u as visited;
6 if $|U| < \omega$ or $\delta(u, v) < \max_{s \in U} \delta(s, v)$ then
7 push u into W, U ;
8 if $|U| > \omega$ then pop U until $|U| = \omega$;
9 return U ;
```

算法 2 类别感知候选邻居选取过程（CatCA）。

## 算法 3. 类别过滤近似最近邻搜索过程 (CFANNS)

输入：查询向量  $q$ ；目标类别  $p$ ；搜索时集束搜索宽度  $\omega_s$ ；目标邻居数量  $k$

输出：查询向量  $q$  的近似  $k$  近邻；且类别均为  $C_t$  或其子类

索引参数：类别映射  $c(\cdot)$ ；类别关系  $T: \leq$

- $ep \leftarrow$  a random vector s.t.  $c(ep) \leq C_t$ ;
- $U \leftarrow \text{CatCA}(q, p, \omega_s, ep)$ ;
- return  $k$  closest vectors to  $q$  in  $U$ ;

算法 3 类别过滤近似最近邻搜索过程（CFANNS）。

## 向 DAG 类别扩展：三邻居压缩

真实场景中类别层次常为 DAG。此时放宽无损压缩约束，改用压缩率更高的三邻居列表：

**N<sub>1</sub>** 仅考虑向量相似度的邻居列表

**N<sub>2</sub>** 祖先类别与后继类别中最相似的向量

**N<sub>3</sub>** 同类别向量中最相似的向量

压缩后构建开销与类别集合  $C$  无关。

## ❺ 实验评估

表 4a 树型类别层次（每个数据集 4 种规模，16 个子数据集）

| 类别数    | 树高 | 平均覆盖类数 |
|--------|----|--------|
| 10     | 5  | 2.80   |
| 100    | 8  | 5.17   |
| 1,000  | 14 | 7.40   |
| 10,000 | 19 | 9.93   |

表 4b 向量数据集

| 数据集      | 规模         | 模态 | 维度    | 距离度量 |
|----------|------------|----|-------|------|
| SIFT     | 1,000,000  | 图像 | 128   | 欧式距离 |
| GIST     | 1,000,000  | 图像 | 960   | 欧式距离 |
| Wikidata | 4,000,000  | 文本 | 1,024 | 余弦距离 |
| Deep10M  | 10,000,000 | 图像 | 96    | 欧式距离 |

- 对比方法：ACORN、Faiss-HNSW、UNG（各自类型中最优）与本文 CatGraph、CatGraph+Comp。DAG 类别：20 / 2,000 / 4,000 类，平均覆盖 5.55 / 48.00 / 493.07。

## ❻ 实验结果 CatGraph 在全部 16 个子数据集上最优：90% 召回率下相比 UNG 提升 5 倍、相比 ACORN 提升 6~10 倍。

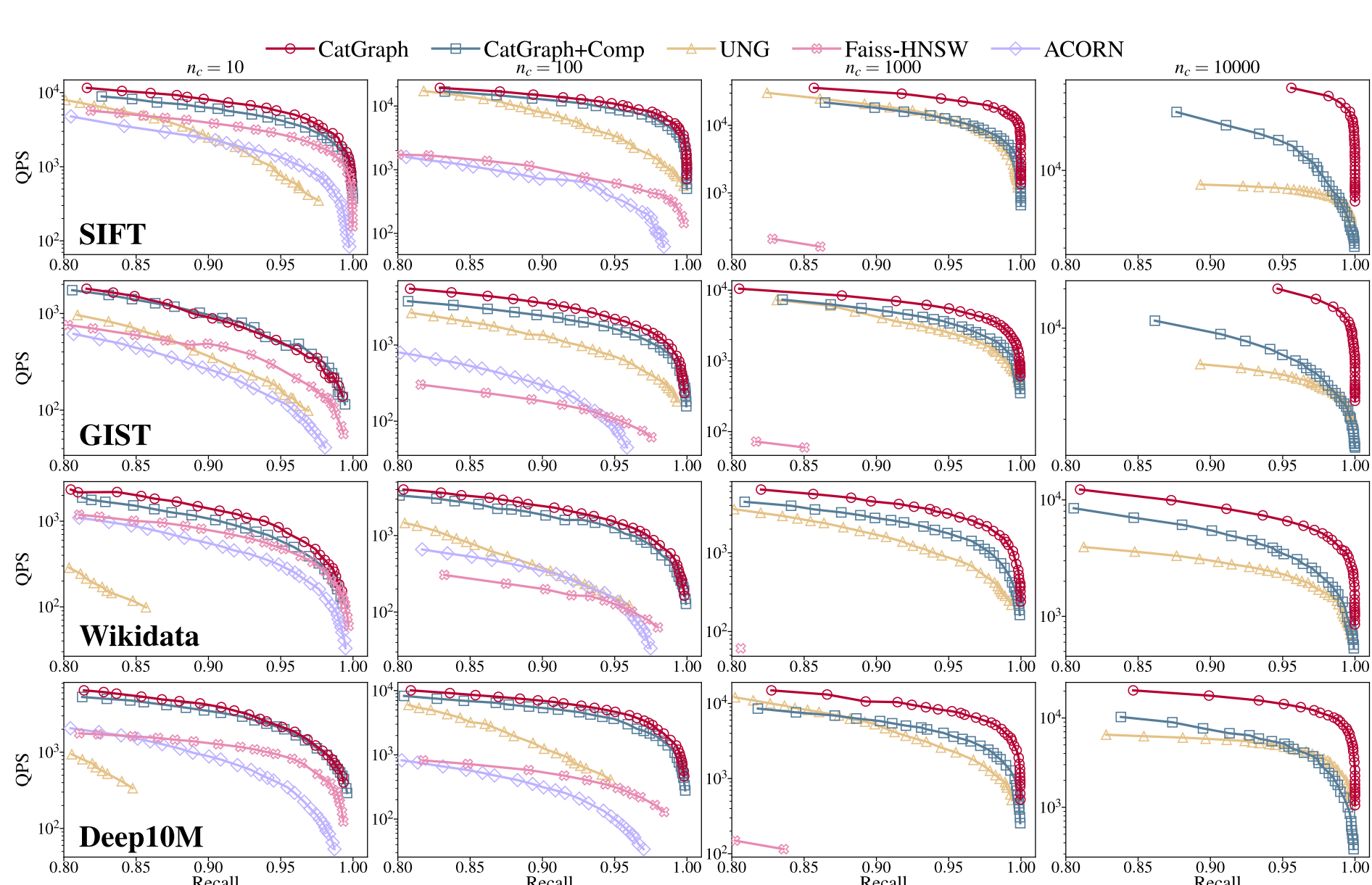
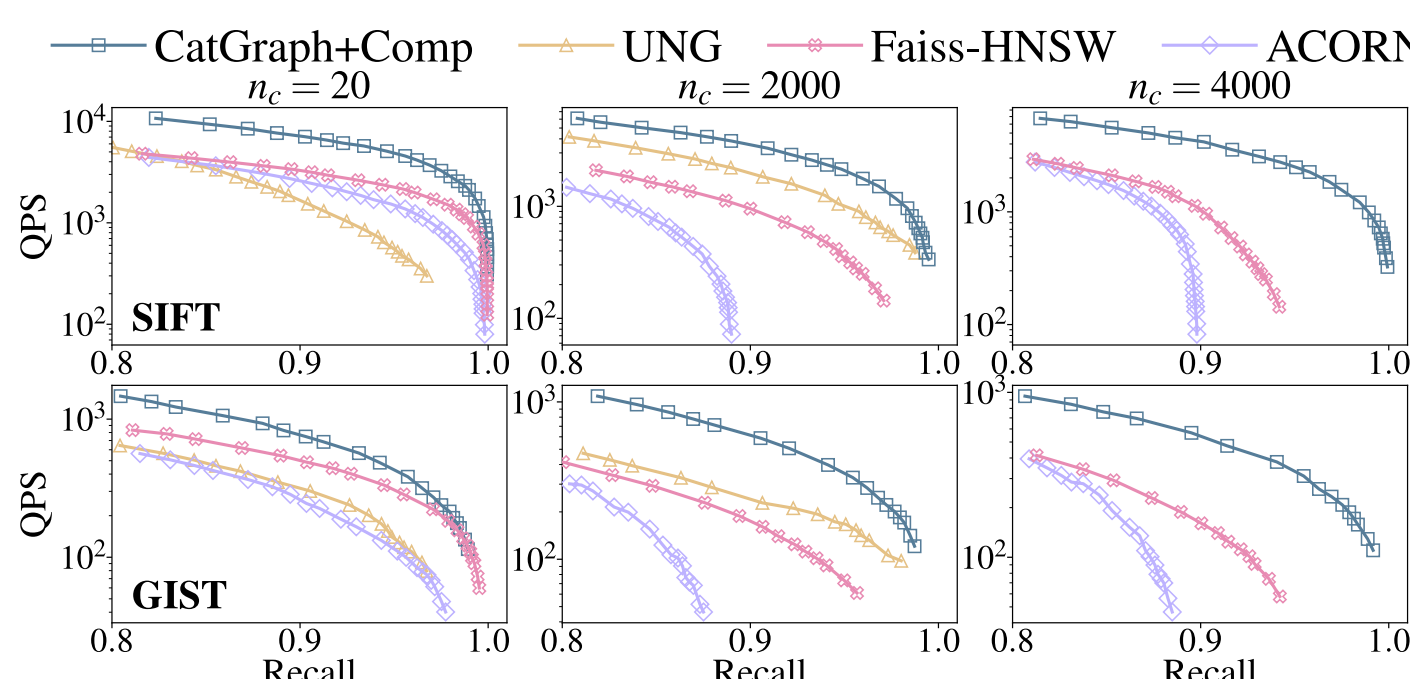
图 3 CFANNS 查询性能：QPS 比较（4 数据集  $\times$  4 类别规模）。

图 4 DAG 类别关系 CFANNS 查询性能。

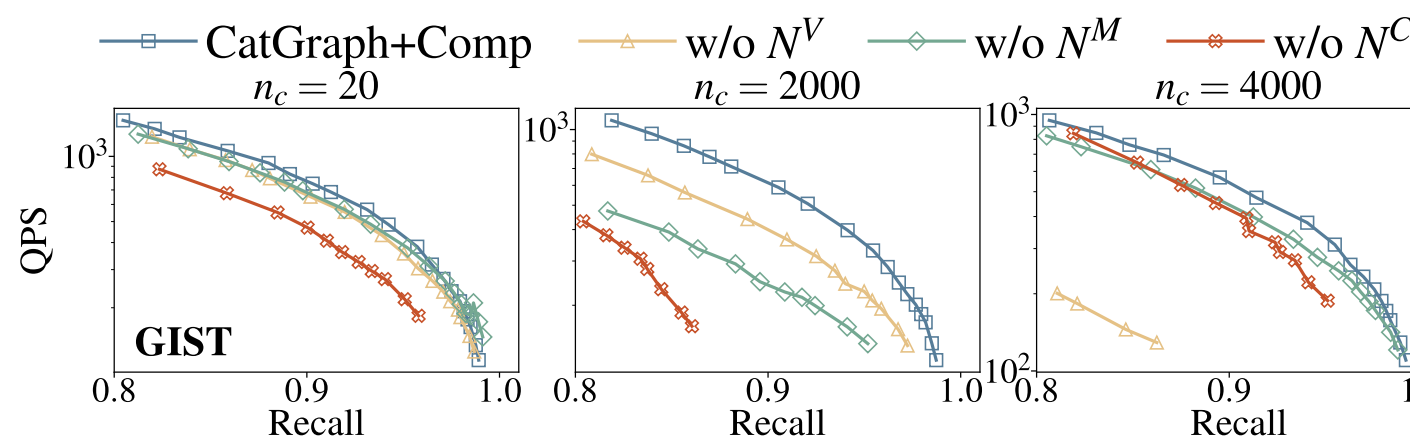


图 5 三类压缩邻居的消融实验。

表 1 索引构建时间（秒）

| 方法           | 10 <sup>1</sup> | 10 <sup>2</sup> | 10 <sup>3</sup> | 10 <sup>4</sup> |
|--------------|-----------------|-----------------|-----------------|-----------------|
| SIFT · ACORN | 103             | 103             | 103             | 103             |
| HNSW         | 30              | 30              | 30              | 30              |
| UNG          | 73              | 26              | 15              | 6               |
| CatGraph     | 80              | 124             | 195             | 844             |
| Comp         | 123             | 105             | 102             | 95              |

Wikidata

| 方法       | 10 <sup>1</sup> | 10 <sup>2</sup> | 10 <sup>3</sup> | 10 <sup>4</sup> |
|----------|-----------------|-----------------|-----------------|-----------------|
| ACORN    | 2,897           | 2,897           | 2,897           | 2,897           |
| HNSW     | 1,250           | 1,250           | 1,250           | 1,250           |
| UNG      | 1,646           | 1,211           | 691             | 234             |
| CatGraph | 861             | 1,568           | 2,221           | 5,389           |
| Comp     | 925             | 1,564           | 1,611           | 1,164           |

n<sub>c</sub> = 10<sup>1</sup>~10<sup>4</sup>；Comp = CatGraph+Comp。

GIST

| 方法       | 10 <sup>1</sup> | 10 <sup>2</sup> | 10 <sup>3</sup> | 10 <sup>4</sup> |
|----------|-----------------|-----------------|-----------------|-----------------|
| ACORN    | 749             | 749             | 749             | 749             |
| HNSW     | 247             | 247             | 247             | 247             |
| UNG      | 481             | 195             | 73              | 14              |
| CatGraph | 365             | 562             | 717             | 1,517           |
| Comp     | 392             | 392             | 355             | 292             |

Deep10M

| 方法       | 10 <sup>1</sup> | 10 <sup>2</sup> | 10 <sup>3</sup> | 10 <sup>4</sup> |
|----------|-----------------|-----------------|-----------------|-----------------|
| ACORN    | 1,350           | 1,350           | 1,350           | 1,350           |
| HNSW     | 392             | 392             | 392             | 392             |
| UNG      | 417             | 221             | 139             | 96              |
| CatGraph | 835             | 1,441           | 2,448           | 8,560           |
| Comp     | 1,335           | 1,443           | 1,400           | 1,292           |

**总结** 提出并形式化 CFANNS 问题；类别增强近邻图将  $O(|C|)$  个先知近邻图无损压缩到  $O(\log |C|)$ 。

**结果** 相比 UNG 查询提升 5 倍、相比 ACORN 提升 6~10 倍；三邻居压缩后构建加速 5~7 倍。

**展望** 面向类别关系可变的动态场景与基于磁盘的索引，支撑亿级向量规模查询。