分类号 TP391???学号17060062

U D C 681???密级公开

工学博士学位论文

基于自然语言模型的文件访问模式分析与文

件预取方法研究

博士生姓名陈辉

学科专业计算机科学与技术

研究方向高性能计算

指导教师周文强研究员

国防科技大学研究生院

二〇一九年十月

An NLP Model-Based Mechanism for

I/O Access Pattern Analysis and File   
Prefetching

Candidate: Hui Chen

Supervisor: Prof. Enqiang Zhou

A dissertation

Submitted in partial fulfillment of the requirements

for the degree of Ph.D of Engineering   
in Computer Science and Technology

Graduate School of National University of Defense Technology   
Changsha, Hunan, P. R. China

October,2019

独创性声明

本人声明所呈交的学位论文是我本人在导师指导下进行的研究工作及取得的

研究成果。尽我所知,除了文中特别加以标注和致谢的地方外,论文中不包含其

他人已经发表和撰写过的研究成果,也不包含为获得国防科技大学或其它教育机

构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献

均已在论文中作了明确的说明并表示谢意。

学位论文题目:基于自然语言模型的文件访问模式分析与文件预取方法研究

学位论文作者签名:日期:年月日

学位论文版权使用授权书

本人完全了解国防科技大学有关保留、使用学位论文的规定。本人授权国防

科技大学可以保留并向国家有关部门或机构送交论文的复印件和电子文档,允许

论文被查阅和借阅;可以将学位论文的全部或部分内容编入有关数据库进行检索,

可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密学位论文在解密后适用本授权书。)

学位论文题目:基于自然语言模型的文件访问模式分析与文件预取方法研究

学位论文作者签名:日期:年月日

作者指导教师签名:日期:年月日

国防科技大学研究生院博士学位论文

目录

摘要. i

ABSTRACT . ii

第一章绪论.1

1.1研究背景.1

1.2国内外研究现状.1

1.3本文主要工作.2

1.4论文结构.2

第二章分布式文件系统缓存预取研究.3

2.1分布式文件系统层次结构.3

2.1.1 GlusterFS 整体架构.3

2.2文件缓存预取技术.3

2.2.1缓存.3

2.2.2主动预取.4

2.3基于 GlusterFS Tiering 功能的缓存管理模块设计.4  
2.3.1 Trace 模块.4

2.3.2访问模式识别模型建立.4

2.3.3运行时缓存管理.4

2.4本章小结.4

第三章基于自然语言处理模型的文件访问模式识别.5

3.1基于词向量模型的文件名向量化.5

3.1.1词向量概述.5

3.1.2基于子词模型的文件名向量化.5

3.2基于循环神经网络的文件访问模式分析.5

3.2.1循环神经网络及其在时间序列分析中的应用.5

3.2.2采用 GRNN(Gated Recurrent Neural Network)的文件访  
问模式分析模型.5

3.3本章小结.5

第四章实验设计与分析.6

4.1实验环境与工作负载.6

4.1.1存储服务器集群搭建.6

4.1.2工作负载选取.6

第 I 页

国防科技大学研究生院博士学位论文

4.2实验数据采集与预处理.6

4.2.1文件名向量化.6

4.2.2文件访问日志记录与预处理.6

4.3循环神经网络模型训练与仿真测试.6

4.4实验结果与分析.6

4.5本章小结.6

第五章总结.7

5.1本文工作总结.7

5.2未来工作展望.7

致谢.8

参考文献.9

附录A 模板提供的希腊字母命令列表.10

第 II 页

国防科技大学研究生院博士学位论文

表目录

第 III 页

国防科技大学研究生院博士学位论文

图目录

第 IV 页

国防科技大学研究生院博士学位论文

摘要

关键词:分布式文件系统;文件预取;词向量;循环神经网络

第 i 页

国防科技大学研究生院博士学位论文

ABSTRACT

Many applications spend a large proportion of the execution time to access �

les. To narrow the increasing gap between computing and I/O performance, several

optimization techniques were adopted, such as data prefetching and data layout

optimization. However, the effectiveness of these optimization processes heavily

depends on the understanding of the I/O behavior. Traditionally, spatial locality

and temporal locality are mainly considered for data prefetching and scheduling

policy. Whereas for most real-world workloads, the � le access pattern is hard to   
capture.

For the goal of deeply and intelligently understanding the I/O access pattern

of modern applications, and efficiently optimizing the performance of current file

systems, we propose a new mechanism to embed file names to vectors and train a

GRNN (Gated Recurrent Neural Network) to provide policies for file prefetching   
and cache replacing.

Key Words: Distributed File Systems, File Prefetching, Word Embed­  
ding, RNN

第 ii 页

国防科技大学研究生院博士学位论文

符号使用说明

HPC 高性能计算(High Performance Computing)

cluster 集群

Itanium 安腾

SMP 对称多处理

API 应用程序编程接口

PI 聚酰亚胺

MPI 聚酰亚胺模型化合物,N-苯基邻苯酰亚胺

PBI 聚苯并咪唑

MPBI 聚苯并咪唑模型化合物,N-苯基苯并咪唑

PY 聚吡咙

PMDA-BDA均苯四酸二酐与联苯四胺合成的聚吡咙薄膜

Δ𝐺活化自由能(Activation Free Energy)

𝜒传输系数(Transmission Coefficient)

𝐸能量

𝑚质量

𝑐光速

𝑃概率

𝑇时间

𝑣速度

第 iii 页

国防科技大学研究生院博士学位论文

1.1研究背景

1.2国内外研究现状

p.14提供性能透明的远程存储的一种方法是将缓存放置在 I / O 堆栈的不同

级别上,以隐藏远程访问的延迟。高速缓存的成功可以通过预取技术来提高,如

前所述,该方法会在应用程序实际请求数据之前尝试从下一级别获取数据,因此

在需要时已存在于高速缓存中。

Eshel 等。(2010年)提出了一种称为"Panache"的缓存文件系统,该系统使

用 pNFS维护 GPFS中存储的数据的分布式缓存。Frings等人提出的技术。(2013)

使用预取来提高使用动态链接库加载并行应用程序的性能。Rajachandrasekar

等。(2013年)提出了一种用户级文件系统,以将检查点请求保留在主内存中,并

将其透明地刷新到持久性存储中。他们的方法包括对远程直接内存访问(RDMA)

的支持。

Zhao 等人提出了另一种缓存中间件。(2014)。他们引入了一种两阶段机制来

减少要在处理节点和中间 I / O 节点之间传输的数据量。Isaila 等。(2011)通过提

出两级预取方案(在客户端和 I / O节点之间,以及 I / O节点和存储服务器之间)

提出了两级预取方案,从而改进了 IBM Blue Gene 的 I / O 转发层。Prabhakar

等。(2010年)通过线性规划对两级缓存系统上的最佳缓存分配进行建模。

坎德米尔等。(2012)定义了请求紧急性的概念,该概念由请求可以延迟多长

时间而不影响应用程序性能给出。它们通过优先处理紧急请求来改进缓存机制。

Seelam 等人的方法。(2010)应用了一个跟踪和检测应用程序访问模式的库。此信

息用于引导预取到本地缓冲区。Patrick 等人提出了类似的方法(使用访问模式检

测指导预取)。(2010),He 等。(2012),Lu 等。(2014年)和 Tang 等人。(2014)。

Suei 等。(2014年)提出了一种使用 SSD 作为 HDD 缓存的存储集群缓存设

计。他们的设计侧重于磨损意识,响应时间和命中率。Zhang 等人也探讨了这个  
想法-快速 SSD 作为 HDD 的缓存。(2012)。

Zhang 等人也使用混合 SSD + HDD 方法。(2013)。他们使用 SSD 来存储

"碎片",即未按条带大小对齐的文件的初始和最后部分。由于获取小部分文件的

性能较低,因此作者认为访问整个文件的性能受到这些碎片的限制,因此,加速

对其的访问可以提高整体性能。Welch 和 Noer(2013)将小文件存储在 SSD 中

以优化对它们的访问,因为他们发现小文件在 PFS 中占多数。He 等人提出的方

法。(2013年)应用成本模型来做出数据放置决策。他们评估文件不同区域的访  
第1页

国防科技大学研究生院博士学位论文

问成本,并将高成本区域放置在 SSD 中。

如第3节所述,这些混合存储解决方案已变得越来越流行,因为仅用固态驱

动器替换所有硬盘将带来高昂的成本。因此,保留 HDD用于存储容量,保留 SSD

用于性能。其他 NVRAM 技术也正在研究中。新的超级计算机有望在计算节点中

包含 NVRAM 设备。这些设备可用作"突发缓冲区",并用于隐藏远程文件系统

的延迟。当前受到关注的研究主题旨在确定如何使用这些突发缓冲区,将其放置

在何处,如何使其透明等(DOE 2014)。刘等。(2012年)评估了在中间 I / O 节  
点中具有突发缓冲区的方法。他们使用模拟进行此分析。

1.3本文主要工作

1.4论文结构

第2页

国防科技大学研究生院博士学位论文

2.1分布式文件系统层次结构

描述主流分布式系统在层次结构上的共性,分层存储带来的优点与伴随的性

能鸿沟。腾讯云社区构建存储系统时需要基于成本和性能来考虑,因此存储系

统通常采用多层不同性价比的存储器件组成存储层次结构。构建高效合理的存储

层次结构,可以在保证系统性能的前提下,降低系统能耗和构建成本,利用数据

访问局部性原理,可以从两个方面对存储层次结构进行优化。从提高性能的角度,

可以通过分析应用特征,识别热点数据并对其进行缓存或预取,通过高效的缓存

预取算法和合理的缓存容量配比,以提高访问性能。从降低成本的角度,采用信息

生命周期管理方法,将访问频率低的冷数据迁移到低速廉价存储设备上,可以在

小幅牺牲系统整体性能的基础上,大幅降低系统的构建成本和能耗。以 GlusterFS   
为例,介绍其层次架构。引用相关数据图表来说明不同层存储之间的性能差异

2.1.1 GlusterFS整体架构

GlusterFS[1]是一个开源、可扩展的分布式文件系统,目前被广泛应用于各类  
商业存储服务器集群,其主要特性如下:

•全局命名空间。

•集群存储管理。

•模块化的层次机构。

•内置 replication and geo-replication 特性。

•自修复功能。

•高效负载均衡。

2.2文件缓存预取技术

接上节内容,引出缓存与预取技术在降低访问延迟方面的巨大作用

2.2.1缓存

被动缓存介绍

第3页

国防科技大学研究生院博士学位论文

2.2.2主动预取

简述主动预取的定义与缓存的区别。

文件预取的定义

预取(prefetching),也称为预分页(prepaging)或预读(read-ahead),是操

作系统数据读取过程中的重要优化方法[2][3][4]。它通过隐藏或减少对非缓存数

据的访问延迟来补充传统的缓存策略(例如 LRU)。其目标是预测将来的数据访  
问,并在请求数据之前使其在内存中可用。

与缓存技术的被动数据迁移不同,预取技术的关键在于对即将访问的数据内

容和生存周期进行主动预测。

•内容预测:与被动缓存不同,主动预取需要预测程序下一阶段可能访问的数

据,是对数据访问的空间局部性的扩充。准确的预测将极大地降低访问延

迟,而错误预测将会引发浪费传输带宽,挤占缓存空间等负面影响。

•时机预测:与缓存机制中的时间局部性类似,主动预取需要针对缓存数据的

时效性和生命周期建立有效的评估。非缓存数据的及时预取有助于提高存命

中率,而清除短期内不再读取的数据将提高缓存空间的利用率。

预取的主要流程

1.针对特定负载,提取访问日志;

2.分析访问日志,对该负载的文件访问模式进行抽象表达;

3.将负载的访问模式作为依据,引导文件系统进行主动预取。

2.3基于 GlusterFS Tiering 功能的缓存管理模块设计  
2.3.1 Trace模块

2.3.2访问模式识别模型建立

2.3.3运行时缓存管理

2.4本章小结

第4页

国防科技大学研究生院博士学位论文

3.1基于词向量模型的文件名向量化

3.1.1词向量概述

3.1.2基于子词模型的文件名向量化

p.1

词向量是将单词在向量空间中分布式表示的自然语言模型,该类模型可帮助

学习算法通过对相似单词进行分组来在自然语言处理任务中实现更好的性能。最

早使用单词表示法的方法可以追溯到1986年 Rumelhart,Hinton 和 Williams 的

工作。此想法此后已成功应用于统计语言建模[1]。后续工作包括自动语音识别和

机器翻译[14、7]以及各种 NLP 任务[2、20、15、3、18、19、9]的应用。

Mikolov 等于2013年引入了 Skip-gram 模型,这是一种从大量非结构化文本  
数据中学习单词的高质量向量表示的有效方法。与大多数以前使用的用于学习

单词向量的神经网络体系结构不同,Skip-gram 模型的训练不涉及密集矩阵乘法,

因此训练过程非常高效:经过优化的单机实现可以在一天中培训超过1000亿个  
单词。

More details about Skip-gram.

p.1通过为每个单词分配不同的向量,学习此类表示形式的流行模型会忽略

单词的形态。这是一个限制,特别是对于具有大量词汇和许多稀有单词的语言。

在本文中,我们提出了一种基于跳过图模型的新方法,其中每个单词都表示为一

包字符 n-gram。向量表示与每个字符 n-gram 相关;单词被表示为这些表示的总

和。我们的方法快速,可以快速在大型语料库上训练模型,并允许我们为未出现

在训练数据中的单词计算单词表示。我们在词的相似性和类比任务上用九种不同

的语言评估词的表示形式。通过与最近提出的形态词表示法进行比较,我们表明,

我们的向量在这些任务上达到了最先进的性能。

3.2基于循环神经网络的文件访问模式分析

3.2.1循环神经网络及其在时间序列分析中的应用

3.2.2采用 GRNN(Gated Recurrent Neural Network)的文件访问模式分  
析模型

3.3本章小结

第5页

国防科技大学研究生院博士学位论文

第四章实验设计与分析

4.1实验环境与工作负载

4.1.1存储服务器集群搭建

4.1.2工作负载选取

编译。网页访问

4.2实验数据采集与预处理

4.2.1文件名向量化

4.2.2文件访问日志记录与预处理

4.3循环神经网络模型训练与仿真测试

4.4实验结果与分析

4.5本章小结

第6页

国防科技大学研究生院博士学位论文

5.1本文工作总结

5.2未来工作展望

第7页

国防科技大学研究生院博士学位论文

致谢

衷心感谢导师 xxx 教授和 xxx 副教授对本人的精心指导。他们的言传身教将  
使我终生受益。

感谢 NudtPaper,它的存在让我的论文写作轻松自在了许多,让我的论文  
格式规整漂亮了许多。

第8页

国防科技大学研究生院博士学位论文

参考文献

[1] GlusterFS: A scalable network filesystem. http://www.gluster.org/.

[2] Griffioen J, Appleton R. Reducing File System Latency using a Predictive

Approach. [C]. In USENIX summer. 1994: 197–207.

[3] Amer A, Long D D, Burns R C. Group-based management of distributed file

caches [C]. In Proceedings 22nd International Conference on Distributed Computing

Systems. 2002: 525–534.

[4] Nanopoulos A, Katsaros D, Manolopoulos Y. A data mining algorithm for

generalized web prefetching [J]. IEEE Transactions on Knowledge & Data En-

gineering. 2003 (5): 1155–1169.

第9页

国防科技大学研究生院博士学位论文

附录A 模板提供的希腊字母命令列表

大写希腊字母:

Γ\Gamma Λ\Lambda Σ\Sigma Ψ\Psi

Δ\Delta Ξ\Xi Υ\Upsilon Ω\Omega

Θ\Theta Π\Pi Φ\Phi

Γ\varGamma Λ\varLambda Σ\varSigma Ψ\varPsi

∆\varDelta Ξ\varXi Υ\varUpsilon Ω\varOmega   
ϴ\varTheta Π\varPi Φ\varPhi

小写希腊字母:

𝛼\alpha 𝜃\theta 𝑜 o 𝜏\tau

𝛽\beta 𝜗\vartheta 𝜋\pi 𝜐\upsilon

𝛾\gamma 𝜄\iota 𝜛\varpi 𝜙\phi

𝛿\delta 𝜅\kappa 𝜌\rho 𝜑\varphi

𝜖\epsilon 𝜆\lambda 𝜚\varrho 𝜒\chi

𝜀\varepsilon 𝜇\mu 𝜎\sigma 𝜓\psi

𝜁\zeta 𝜈\nu 𝜍\varsigma 𝜔\omega

𝜂\eta 𝜉\xi 𝜘\varkappa \digamma

α\upalpha θ\uptheta o \mathrm{o}τ\uptau

β\upbeta ϑ\upvartheta π\uppi υ\upupsilon

γ\upgamma ι\upiota ϖ\upvarpi ϕ\upphi

δ\updelta κ\upkappa ρ\uprho φ\upvarphi

ϵ\upepsilon λ\uplambda ϱ\upvarrho χ\upchi

ε\upvarepsilon μ\upmu σ\upsigma ψ\uppsi

ζ\upzeta ν\upnu ς\upvarsigma ω\upomega   
η\upeta ξ\upxi

希腊字母属于数学符号类别,请用\bm命令加粗,其余向量、矩阵可

用\mathbf。

第10页