



(12)发明专利申请

(10)申请公布号 CN 108038049 A

(43)申请公布日 2018.05.15

(21)申请号 201711333074.7

(22)申请日 2017.12.13

(71)申请人 西安电子科技大学

地址 710071 陕西省西安市太白南路2号西安电子科技大学

(72)发明人 裴庆祺 赵伟伟 王磊

(74)专利代理机构 西安长和专利代理有限公司 61227

代理人 黄伟洪

(51) Int. Cl.

G06F 11/34(2006.01)

权利要求书3页 说明书8页 附图3页

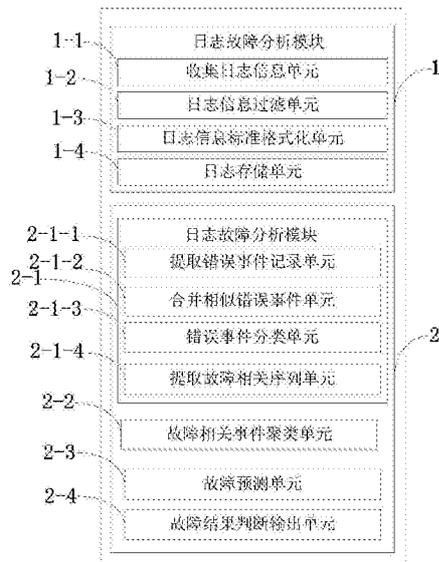
(54)发明名称

实时日志控制系统及控制方法、云计算系统及服务器

(57)摘要

本发明属于云计算技术领域,公开了一种实时日志控制系统及控制方法、云计算系统及服务器,通过对于日志记录事件的分析,将错误信息进行分类、过滤、聚合操作,提取成为序列,训练故障模型并计算序列属于故障序列的概率和非故障序列的概率,使用贝叶斯分类理论得出结果,做出预测。本发明通过对于日志记录事件的分析,将所有的错误信息进行分类、过滤、聚合等操作,提取成为序列,训练故障模型并计算该序列属于故障序列的概率和非故障序列的概率,使用贝叶斯分类理论得出结果,做出预测,比起大量的规则匹配来说提高了判断速度;故障预测研究对于减轻网络管理和维护的负担,减少网络故障造成的损失具有重要意义。

CN 108038049 A



1. 一种实时日志控制方法,其特征在于,所述实时日志控制方法通过对于日志记录事件的分析,将错误信息进行分类、过滤、聚合操作,提取成为序列,训练故障模型并计算序列属于故障序列的概率和非故障序列的概率,使用贝叶斯分类理论得出结果,做出预测。

2. 如权利要求1所述的实时日志控制方法,其特征在于,所述实时日志控制方法具体包括:

步骤一,收集分布式系统中各个节点上的日志文件数据,通过增量检查将新产生日志数据实时地发送给收集端;

步骤二,删除在某一时间段内相同位置报告的相同类型事件,删除冗余事件,通过设置时间阈值 $\Delta t = \Delta s$ 表示用于执行事件过滤的时间窗口;通过移除某一时间段内由多个不同位置报告的相似事件,删除日志中的冗余事件,将数据流保存到时序数据库中;使用相似性 $\text{Sim}(D_1, D_2)$ 来判断:

$$\text{Sim}(D_1, D_2) = \cos \theta = \frac{\sum_{k=1}^n W_{1k} \times W_{2k}}{\sqrt{(\sum_{k=1}^n W_{1k})^2 \times (\sum_{k=1}^n W_{2k})^2}}$$

其中 D_1, D_2 表示两个序列, W_{1k}, W_{2k} 表示 D_1, D_2 序列的向量项,相似度即两个向量夹角的余弦值来表示, $\text{Sim}(D_1, D_2)$ 越大,表示两者相似度高;

步骤三,在每条数据存储到数据表时,利用SQL语句按照时间戳、进程号、记录级别、进程模块、分隔符、记录信息分割记录;

步骤四,利用SQL语句将处理过的标准格式化数据进行持久化存储;

步骤五,提取日志故障序列;

步骤六,聚类标准根据序列的似然值 $P(o|x)$ 作为度量值来计算,采用层次聚类算法实现故障相关事件分组,其中:

$$P(o|x) = \sum_{\pi} \pi_{s_0} b_{s_0}(o_0) \prod_{k=1}^L P(S_k = s_k, d_k = t_k - t_{k-1} | S_{k-1} = s_{k-1}) b_{s_k}(o_k);$$

$S = [s_i]$ 表示一个长为 L 状态序列, $b_{s_i}(o_i)$ 为在状态 $s_i(k)$ 在初始状态概率向量 $\pi = [\pi_i]$ 下的观测值的概率矩阵;

步骤七,采用隐半马尔可夫模型HSMM和贝叶斯网络Bayes Net相结合,对实时日志数据做出故障预测;

标准HSMM可由状态之间转化概率矩阵 $G(t) = [g_{ij}(t)]$ 、状态 $s_i(k)$ 在初始状态概率向量 $\pi = [\pi_i]$ 下的观测值的概率矩阵 $B = b_i(k)$, 定义为 $\lambda = (\pi, G(t), B)$; 将状态持续时间概率分布连续化;将状态持续时间的分布作为连续分布来处理,并且假设其服从威布尔分布来描述状态持续时间概率分布,状态的状态持续时间概率分布 $f_i(l)$ 为:

$$f_i(l) = \alpha \beta (\alpha l)^{\beta-1} e^{-(\alpha l)^\beta};$$

式中: α, β 分别为威布尔分布的尺度参数和形状参数;

将状态监测值概率分布连续化;同样设定其服从威布尔分布,状态检测值概率分布函数 $\xi_i(\theta)$ 为:

$$\xi_i(\theta) = P(y_k = \theta | q_k = x_k) = \alpha_i \beta_i (\alpha_i \theta)^{\beta_i-1} e^{-(\alpha_i \theta)^{\beta_i}};$$

其中 α_i, β_i 为各状态阶段的威布尔分布的参数;改进的 HSMM 模型可描述为 $\lambda = (\pi, f_i(t), \xi_i)$;

步骤八,故障和非故障模型进行训练,参数 λ_F 和 $\lambda_{\bar{F}}$,目标是评估,给定一个观察序列 $O = [o_1, o_2, \dots, o_n]$ 是否为故障相关序列;计算分类模型的序列似然值,随后被分类为无故障或故障贝叶斯决策理论;

步骤九,故障结果预判:

$$\log[P(o|\lambda_F)] - \log[P(o|\lambda_{\bar{F}})] > \log\left[\frac{C_{FF} - C_{\bar{F}\bar{F}}}{C_{F\bar{F}} - C_{\bar{F}F}}\right] + \log\left[\frac{P(F)}{P(\bar{F})}\right];$$

将一个序列标记成为故障相关事件序列,系统发出故障预测;其中 C_{FF} 表示错误的将故障相关序列判断成为故障无关序列的代价, $P(F)$ 表示故障的概率, $\log[P(o|\lambda_F)]$ 表示对序列似然值取对数。

3. 如权利要求2所述的实时日志控制方法,其特征在于,所述提取日志故障序列具体包括:

第一步,提取错误事件序列:利用SQL语句,根据日志等级将ERROR级别的记录过提取出来,保留时间戳和文本消息信息;

第二步,合并相似错误事件:对事件序列利用Levenshtein编辑距离算法,将相似度较大的错误事件合并;最小编辑距离包含子最小编辑距离;

$$\begin{cases} 0 & i=0 \text{ 或 } j=0 \\ \min(d_{i-1,j} + 1, d_{i,j-1} + 1, d_{i-1,j-1}) & x_i = y_j \\ \min(d_{i-1,j} + 1, d_{i,j-1} + 1, d_{i-1,j-1} + 1) & x_i \neq y_j; \end{cases}$$

其中 $d_{[i-1,j]}+1$ 代表目标日志插入一个字母, $d_{[i,j-1]}+1$ 代表匹配日志删除一个字母;然后 $x_i = y_j$ 时,不需要修改,所以和上一步 $d_{[i-1,j-1]}+1$ 代价相同,否则 $+1$, $d_{[i,j]}$ 表示以上三者中最小的一项;

第三步,错误事件分类:经过上一步将错误事件合并后,根据错误事件的文本信息中的关键字将相似的错误事件进行归类,并赋值ID,保存在数据库中;

第四步,提取序列:按照时间顺序,提取在故障发生前一段时间 $\Delta t_{\bar{F}}$ 内的事件,设定为故障相关事件序列, $\Delta t_{\bar{F}}$ 为故障前置时间,当前故障事件为相关故障事件;非故障相关事件序列则是在系统未发生故障的时间区间内的事件序列。

4. 一种如权利要求1所述实时日志控制方法的实时日志控制系统,其特征在于,所述实时日志控制系统包括:日志信息处理模块、日志故障分析模块。

5. 如权利要求4所述的实时日志控制系统,其特征在于,所述日志故障分析模块包括:

收集日志信息单元,用于收集分布式系统中各个节点上的日志文件数据,日志收集功能应该允许自定义所要监听的日志文件,通过增量检查的方法,将新产生日志数据实时地发送给收集端;

日志信息过滤单元,用于进行数据的去冗余和过滤;

日志信息标准格式化单元,用于处理过的日志信息进行数据标准格式化;

日志存储单元,用于将处理过的标准格式化数据进行持久化存储。

6. 如权利要求4所述的实时日志控制系统,其特征在于,所述日志故障分析模块包括:
 - 提取日志事件序列单元;
 - 故障相关事件聚类单元,用于利用事件提前训练出一个小的隐半马尔可夫模型,求序列似然值;
 - 故障预测单元,使用隐半马尔可夫模型和贝叶斯分贝理论,判定序列是否为故障相关序列;
 - 故障结果判断输出单元:当判定为故障相关序列时,系统发出故障警告流,输出状态故障预警。
7. 如权利要求6所述的实时日志控制系统,其特征在于,所述提取日志事件序列单元进一步包括:
 - 提取错误事件记录单元,根据日志等级将ERROR级别的记录过提取出来,保留时间戳、进程模块和文本消息信息;
 - 合并相似错误事件单元,将错误事件序列利用Levenshtein编辑距离算法,将相似度较大的错误事件合并;
 - 错误事件分类单元,对事件序列采用Levenshtein编辑距离算法,将相似的错误事件进行归类,并赋值ID;
 - 提取故障相关序列单元,按照时间先后顺序,提取故障前一段时间内的事件,设定为故障前置事件。
8. 一种利用权利要求1~3任意一项所述实时日志控制方法的云计算系统。
9. 一种利用权利要求1~3任意一项所述实时日志控制方法的云计算服务器。

实时日志控制系统及控制方法、云计算系统及服务器

技术领域

[0001] 本发明属于云计算技术领域,尤其涉及一种实时日志控制系统及控制方法、云计算系统及服务器。

背景技术

[0002] 随着计算机技术的高速发展,云计算成为最重要的计算机领域之一,云计算服务深入到每个人的生活和工作当中。能够通过对实时数据的计算,基于机器学习算法对于云计算系统中可能发生的故障进行提前预测,预留出故障响应时间,同时还支持弹性地平扩展集群的处理能力,以适应不断增长的数据量和用户需求。对海量日志数据进行实时计算处理,从数据中挖掘分析出系统的状态、故障预测方面具有良好的发展方向和应用前景。

[0003] 综上所述,现有技术存在的问题是:原有的故障预测模型中,一方面,状态持续时间分布大多默认为指数型分布,而实际中故障的状态概率变化并不满足指数型;另一方面,在故障状态检测值概率做了离散化处理,这对大数据环境进行实验分析会有意料之外的影响,故本内容采用状态持续时间分布和状态观察值概率分布进行连续化分布即假定威布尔分布,采用改进的预测模型可提高诊断和预测的概率值。

发明内容

[0004] 针对现有技术存在的问题,本发明提供了一种实时日志控制系统及控制方法、云计算系统及服务器。

[0005] 本发明是这样实现的,一种实时日志控制方法,所述实时日志控制方法通过对于日志记录事件的分析,将错误信息进行分类、过滤、聚合操作,提取成为序列,训练故障模型并计算序列属于故障序列的概率和非故障序列的概率,使用贝叶斯分类理论得出结果,做出预测。

[0006] 进一步,所述实时日志控制方法具体包括:

[0007] 步骤一,收集分布式系统中各个节点上的日志文件数据,通过增量检查将新产生日志数据实时地发送给收集端;

[0008] 步骤二,删除在某一时间段内相同位置报告的相同类型事件,删除冗余事件,通过设置时间阈值 $\Delta t = \Delta s$ 表示用于执行事件过滤的时间窗口;通过移除某一时间段内由多个不同位置报告的相似事件,删除日志中的冗余事件,将数据流保存到时序数据库中;使用相似性 $\text{Sim}(D_1, D_2)$ 来判断:

$$[0009] \quad \text{Sim}(D_1, D_2) = \cos \theta = \frac{\sum_{k=1}^n W_{1k} \times W_{2k}}{\sqrt{(\sum_{k=1}^n W_{1k})^2 \times (\sum_{k=1}^n W_{2k})^2}};$$

[0010] 其中 D_1, D_2 表示两个序列, W_{1k}, W_{2k} 表示 D_1, D_2 序列的向量项,相似度即两个向量夹角的余弦值来表示, $\text{Sim}(D_1, D_2)$ 越大,表示两者相似度越高;

[0011] 步骤三,在每条数据存储到数据表时,利用SQL语句按照时间戳、进程号、记录级别、进程模块、分隔符、记录信息分割记录;

[0012] 步骤四,利用SQL语句将处理过的标准格式化数据进行持久化存储;

[0013] 步骤五,提取日志故障序列;

[0014] 步骤六,聚类标准根据序列的似然值 $P(o|\lambda)$ 作为度量值来计算,采用层次聚类算法实现故障相关事件分组,其中:

$$P(o|\lambda) =$$

$$\sum \pi_{s_0} b_{s_0}(o_0) \prod_{x=1}^L P(S_x = s_x, d_x = t_x - t_{x-1} | S_{x-1} = s_{x-1}) b_{s_x}(o_x);$$

[0016] $S = [s_i]$ 表示一个长为L状态序列, $b_{s_i}(o_i)$ 为在状态 $s_i(k)$ 在初始状态概率向量 $\pi = [\pi_i]$ 下的观测值的概率矩阵;

[0017] 步骤七,采用改进的HSMM和贝叶斯网络BayesNet相结合,对实时日志数据做出故障预测;

[0018] 标准HSMM可由状态之间转化概率矩阵 $G(t) = [g_{ij}(t)]$ 、状态 $s_i(k)$ 在初始状态概率向量 $\pi = [\pi_i]$ 下的观测值的概率矩阵 $B = b_i(k)$,定义为 $\lambda = (\pi, G(t), B)$;将状态持续时间概率分布连续化;将状态持续时间的分布作为连续分布来处理,并且假设其服从威布尔分布来描述状态持续时间概率分布,状态的状态持续时间概率分布 $f_i(l)$ 为:

$$f_i(l) = \alpha\beta(\alpha l)^{\beta-1} e^{-(\alpha l)^\beta};$$

[0020] 式中: α 、 β 分别为威布尔分布的尺度参数和形状参数;

[0021] 将状态监测值概率分布连续化;同样设定其服从威布尔分布,状态检测值概率分布函数 $\xi_i(\theta)$ 为:

$$\xi_i(\theta) = P(x_t = \theta | q_t = x_t) = \alpha_i \beta_i (\alpha_i \theta)^{\beta_i - 1} e^{-(\alpha_i \theta)^{\beta_i}};$$

[0023] 其中 α_i 、 β_i 为各状态阶段的威布尔分布的参数;改进的HSMM模型可描述为 $\lambda = (\pi, f_i(l), \xi_i)$;

[0024] 步骤八,故障和非故障模型进行训练,参数 $\lambda_{\bar{F}}$ 和 λ_F ,目标是评估,给定一个观察序列 $O = [o_1, o_2, \dots, o_L]$ 是否为故障相关序列;计算分类模型的序列似然值,随后被分类为无故障或故障贝叶斯决策理论;

[0025] 步骤九,故障结果预判:

$$\log[P(o|\lambda_{\bar{F}})] - \log[P(o|\lambda_F)] > \log \left[\frac{C_{\bar{F}\bar{F}} - C_{\bar{F}F}}{C_{F\bar{F}} - C_{FF}} \right] + \log \left[\frac{P(\bar{F})}{P(F)} \right];$$

[0027] 将一个序列标记成为故障相关事件序列,系统发出故障预测;其中 $C_{\bar{F}\bar{F}}$ 表示错误的将故障相关序列判断成为故障无关序列的代价, $P(\bar{F})$ 表示故障的概率, $\log[P(o|\lambda_{\bar{F}})]$ 表示对序列似然值取对数。

[0028] 进一步,所述提取日志故障序列具体包括:

[0029] 第一步,提取错误事件序列:利用SQL语句,根据日志等级将ERROR级别的记录过提取出来,保留时间戳和文本消息信息;

[0030] 第二步,合并相似错误事件:对事件序列利用Levenshtein编辑距离算法,将相似程度较大的错误事件合并;最小编辑距离包含于最小编辑距离;

$$[0031] \quad \begin{cases} 0 & i=0 \text{ 或 } j=0 \\ \min(d_{i-1,j}+1, d_{i,j-1}+1, d_{i-1,j-1}) & x_i = y_j \\ \min(d_{i-1,j}+1, d_{i,j-1}+1, d_{i-1,j-1}+1) & x_i \neq y_j; \end{cases}$$

[0032] 其中 $d_{[i-1,j]+1}$ 代表目标日志插入一个字母, $d_{[i,j-1]-1}$ 代表匹配日志删除一个字母;然后 $x_i = y_j$ 时,不需要修改,所以和上一步 $d_{[i-1,j-1]+1}$ 代价相同,否则+1, $d_{[i,j]}$ 表示以上三者中最小的一项;

[0033] 第三步,错误事件分类:经过上一步将错误事件合并后,根据错误事件的文本信息中的关键字将相似的错误事件进行归类,并赋值ID,保存在数据库中;

[0034] 第四步,提取序列:按照时间顺序,提取在故障发生前一段时间 Δt_x 内的事件,设定为故障相关事件序列, Δt_x 为故障前置时间,当前故障事件为相关故障事件;非故障相关事件序列则是在系统未发生故障的时间区间内的事件序列。

[0035] 本发明的另一目的在于提供一种所述实时日志控制方法的实时日志控制系统,所述实时日志控制系统包括:日志信息处理模块、日志故障分析模块。

[0036] 进一步,所述日志故障分析模块包括:

[0037] 收集日志信息单元,用于收集分布式系统中各个节点上的日志文件数据,日志收集功能应该允许自定义所要监听的日志文件,通过增量检查的方法,将新产生日志数据实时地发送给收集端;

[0038] 日志信息过滤单元,用于进行数据的去冗余和过滤;

[0039] 日志信息标准格式化单元,用于处理过的日志信息进行数据标准格式化;

[0040] 日志存储单元,用于将处理过的标准格式化数据进行持久化存储。

[0041] 进一步,所述日志故障分析模块包括:

[0042] 提取日志事件序列单元;

[0043] 故障相关事件聚类单元,用于利用事件提前训练出一个小的隐半马尔可夫模型,求序列似然值;

[0044] 故障预测单元,使用隐半马尔可夫模型和贝叶斯分贝理论,判定序列是否为故障相关序列;

[0045] 故障结果判断输出单元:当判定为故障相关序列时,系统发出故障警告流,输出状态故障预警。

[0046] 所述提取日志事件序列单元进一步包括:

[0047] 提取错误事件记录单元,根据日志等级将ERROR级别的记录过提取出来,保留时间戳、进程模块和文本消息信息;

[0048] 合并相似错误事件单元,将错误事件序列利用Levenshtein编辑距离算法,将相似度较大的错误事件合并;

[0049] 错误事件分类单元,对事件序列采用Levenshtein编辑距离算法,将相似的错误事件进行归类,并赋值ID;

[0050] 提取故障相关序列单元,按照时间先后顺序,提取故障前一段时间内的事件,设定为故障前置事件。

[0051] 本发明的另一目的在于提供一种利用所述实时日志控制方法的云计算系统。

[0052] 现今故障预测研究工作主要有三类方法,包括基于日志频率的故障检测模型,基于消息频率的故障检测模型和基于状态转移的故障检测模型。

[0053] 本发明在系统运行时间内实时收集日志信息并进行聚类处理,通过分析事件日志使用机器学习的算法和模型,实现对系统未来可能发生的故障的预测,在系统运行过程中对系统故障进行提前排查和定位,用于提高系统运维效率和预防紧急故障事件。本发明通过对于日志记录事件的分析,将所有的错误信息进行分类、过滤、聚合等操作,提取成为序列,训练故障模型并计算该序列属于故障序列的概率和非故障序列的概率,使用贝叶斯分类理论得出结果,做出预测。

[0054] 该方法的有效判断标准主要由三个参数来决定,即准确率、召回率以及F-measure参数,准确率反应的是所有预测中正确的比率,召回率反应的是所有故障中被正确预测出来的比率,F.measure是结合准确率和召回率的一个综合衡量值;

[0055] 预测情况如下表1:

[0056]

预测结果\实际结果	系统故障	系统正常
系统故障	TruePositive (TP)	FalsePositive (FP)
系统正常	FalseNegative (FN)	TrueNegative (TN)

[0057] 表1预测情况

[0058] 预测有效性参数如表2:

[0059]

标准参数	计算参数表达式
正确率(P)	$P = \frac{TP}{FN + FP}$
召回率 R	$R = \frac{TP}{TP + FN}$
F-measure(F)	$F = \frac{2 * P * R}{P + R}$

[0060] 表2有效性参数表达式

[0061] 经过系统实验得出下面数据结论,可看出本次系统在准确率上优于未改进之前

[0062]

故障预测方法	准确率(P)	召回率(R)	F-measure(F)
DFT	0.314	0.458	0.373
HSMM	0.535	0.439	0.482
改进 HSMM	0.852	0.657	0.741

附图说明

[0063] 图1是本发明实施例提供的实时日志控制系统结构示意图;

[0064] 图中:1、日志信息处理模块;1-1、收集日志信息单元;1-2、日志信息过滤单元;1-

3、日志信息标准格式化单元；1-4、日志存储单元；2、日志故障分析模块；2-1、提取日志事件序列单元；2-1-1、提取错误事件记录单元；2-1-2、合并相似错误事件单元；2-1-3、错误事件分类单元；2-1-4、提取故障相关序列单元；2-2、故障相关事件聚类单元；2-3、故障预测单元；2-4、故障结果判断输出单元。

[0065] 图2是本发明实施例提供的实时日志控制方法流程图。

[0066] 图3是本发明实施例提供的实时日志控制方法的实现流程图。

[0067] 图4是本发明实施例提供的故障序列提取示意图。

具体实施方式

[0068] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[0069] 下面结合附图对本发明的应用原理作详细的描述。

[0070] 如图1所示，本发明实施例提供的实时日志控制系统包括：日志信息处理模块1、日志故障分析模块2。

[0071] 日志故障分析模块1包括：

[0072] 收集日志信息单元1-1：用于收集分布式系统中各个节点上的日志文件数据，日志收集功能应该允许自定义所要监听的日志文件，通过增量检查的方法，将新产生日志数据实时地发送给收集端。

[0073] 日志信息过滤单元1-2：用于进行数据的去冗余和过滤。

[0074] 日志信息标准格式化单元1-3：用于处理过的日志信息进行数据标准格式化，比如按照：时间戳、进程号、记录级别、进程模块、分隔符、记录信息，其中，记录级别分为几大类，包括：ERROR、WARNING、TRACE、INFO、DEBUG、CRITICAL、AUDIT，级别越靠前等级越高，等级越高代表事件的重要程度越高。

[0075] 日志存储单元1-4：用于将处理过的标准格式化数据进行持久化存储，便于后期数据的提取分析。

[0076] 日志故障分析模块2包括：

[0077] 提取日志事件序列单元2-1：

[0078] 故障相关事件聚类单元2-2，用于利用事件提前训练出一个小的隐半马尔可夫(HSMM)模型，求序列似然值即给定序列利用训练模型产生的观察序列；

[0079] 故障预测单元2-3：使用隐半马尔可夫模型和贝叶斯分贝理论，判定序列是否为故障相关序列；

[0080] 故障结果判断输出单元2-4：当判定为故障相关序列时，系统发出故障警告流，输出状态故障预警。

[0081] 提取日志事件序列单元2-1进一步包括：

[0082] 提取错误事件记录单元2-1-1：根据日志等级将ERROR级别的记录过提取出来，保留时间戳、进程模块和文本消息等信息；

[0083] 合并相似错误事件单元2-1-2：将错误事件序列利用Levenshtein编辑距离算法，将相似度较大的错误事件合并；

[0084] 错误事件分类单元2-1-3:对事件序列采用Levenshtein编辑距离算法,将相似的错误事件进行归类,并赋值ID;

[0085] 提取故障相关序列单元2-1-4:按照时间先后顺序,提取故障前一段时间内的事件,设定为故障前置事件。

[0086] 如图2所示,本发明实施例提供的实时日志控制方法包括以下步骤:

[0087] S201:通过对于日志记录事件的分析,将所有的错误信息进行分类、过滤、聚合等操作,提取成为序列;

[0088] S202:训练故障模型并计算该序列属于故障序列的概率和非故障序列的概率,使用贝叶斯分类理论得出结果,做出预测。

[0089] 下面结合附图对本发明的应用原理作进一步的描述。

[0090] 比起利用故障关键字进行大量的规则匹配来说,在本发明中,采用改进的HSMM(隐马尔科夫模型)和Bayesdecisiontheory(贝叶斯分类理论),直接计算一个错误序列属于故障序列的概率,提高判断速度。

[0091] 如图3所示,本发明实施例提供的实时日志控制方法具体步骤如下:

[0092] 1、日志信息处理过程

[0093] 步骤1,日志信息收集

[0094] 系统应该能够收集分布式系统中各个节点上的日志文件数据,日志收集功能应该允许自定义所要监听的日志文件,通过增量检查的方法,即将新产生日志数据实时地发送给收集端。

[0095] 步骤2,日志信息过滤

[0096] 有两种方法:一个是时间过滤,另一个是空间过滤。当系统检测到异常时,在系统发生故障之前,系统会持续输出警告信息流。同样地,一旦系统发生故障,在解决故障问题之前日志中可能会多次反复出现故障信息。

[0097] 时间过滤方法通过删除在某一时间段内相同位置报告的相同类型事件,从而删除冗余事件,通过设置时间阈值 $\Delta t = 1s$ 表示用于执行事件过滤的时间窗口。空间过滤方法通过移除某一时间段内由多个不同位置报告的相似事件,删除日志中的冗余事件,将数据流保存到时序数据库中,节省空间并提高效率。通常使用相似性 $Sim(D_1, D_2)$ 来判断:

$$[0098] \quad Sim(D_1, D_2) = \cos \theta = \frac{\sum_{k=1}^n W_{1k} \times W_{2k}}{\sqrt{(\sum_{k=1}^n W_{1k})^2 \times (\sum_{k=1}^n W_{2k})^2}}$$

[0099] 其中 D_1, D_2 表示两个序列, W_{1k}, W_{2k} 表示 D_1, D_2 序列的向量项,相似度即两个向量夹角的余弦值来表示, $Sim(D_1, D_2)$ 越大,表示两者相似度越高。

[0100] 步骤3,日志格式标准化。

[0101] 在将每条数据存储到数据表时,利用SQL语句按照时间戳、进程号、记录级别、进程模块、分隔符、记录信息等分割记录。

[0102] 步骤4,日志存储。

[0103] 利用SQL语句将处理过的标准格式化数据进行持久化存储,便于后期数据的提取分析。

[0104] 2.日志故障分析:

[0105] 在故障表现和系统状态之间建立基于概率因果关系,通过故障出现的先验概率来

对隐半马尔科夫模型和贝叶斯网络进行训练,诊断时根据先验概率求解故障表现下各种系统状态的后验概率,直观表达变量的联合概率分布,同时计算各特征造成故障的概率。

[0106] 步骤1,提取日志故障序列。

[0107] 第一步,提取错误事件序列:利用SQL语句,根据日志等级将ERROR级别的记录过提取出来,保留时间戳和文本消息等信息;

[0108] 第二步,合并相似错误事件:对上一步骤的事件序列利用Levenshtein编辑距离算法,将相似度较大的错误事件合并;

[0109] 该算法使用了动态规划的算法策略,该问题具备最优子结构,最小编辑距离包含子最小编辑距离;

$$[0110] \begin{cases} 0 & i=0 \text{ 或 } j=0 \\ \min(d_{i-1,j} + 1, d_{i,j-1} + 1, d_{i-1,j-1}) & x_i = y_j \\ \min(d_{i-1,j} + 1, d_{i,j-1} + 1, d_{i-1,j-1} + 1) & x_i \neq y_j; \end{cases}$$

[0111] 其中 $d_{[i-1,j]+1}$ 代表目标日志插入一个字母, $d_{[i,j-1]+1}$ 代表匹配日志删除一个字母;然后 $x_i = y_j$ 时,不需要修改,所以和上一步 $d_{[i-1,j-1]+1}$ 代价相同,否则+1, $d_{[i,j]}$ 表示以上三者中最小的一项;

[0112] 第三步,错误事件分类:经过上一步将错误事件合并后,根据错误事件的文本信息中的关键字将相似的错误事件进行归类,并赋值ID,保存在数据库中;

[0113] 第四步,提取序列:按照时间顺序,提取在故障发生前一段时间 $\forall t_x$ 为的事件,设定为故障相关事件序列, $\forall t_x$ 为故障前置时间,当前故障事件为相关故障事件;非故障相关事件序列则是在系统未发生故障的时间区间内的事件序列,如图4所示:

[0114] 步骤2,故障相关事件聚类。

[0115] 实际中,会有多种的故障相关事件序列可能导致同一种的系统故障,而这多种故障相关事件序列的特征是不同的,故需要进行聚类。

[0116] 聚类标准可根据序列的似然值 $P(o|\lambda)$ 作为度量值来计算,最后采用层次聚类算法实现故障相关事件分组,其中:

$$[0117] P(o|\lambda) = \sum_{\pi_0} b_{\pi_0}(o_0) \prod_{k=1}^L P(S_k = s_k, d_k = t_k - t_{k-1} | S_{k-1} = s_{k-1}) b_{\pi_k}(o_k);$$

[0118] $S = [s_i]$ 表示一个长为L状态序列, $b_{s_i}(o_i)$ 为在状态 $s_i(k)$ 在初始状态概率向量 $\pi = [\pi_i]$ 下的观测值的概率矩阵。

[0119] 步骤3,训练建立预测模型。

[0120] 预测模型是网络故障预测的关键,所构造的特征直接影响预测模型的性能。本次采用隐半马尔可夫模型(HSMM)和贝叶斯网络(Bayes Net)相结合,针对实时日志数据做出故障预测。

[0121] 标准HSMM可由状态之间转化概率矩阵 $G(t) = [g_{ij}(t)]$ 、状态 $s_i(k)$ 在初始状态概率向量 $\pi = [\pi_i]$ 下的观测值的概率矩阵 $B = b_i(k)$,定义为 $\lambda = (\pi, G(t), B)$ 。

[0122] 本次对HSMM的改进方面有:将状态持续时间概率分布连续化。将状态持续时间的分布作为连续分布来处理,并且假设其服从威布尔分布来描述状态持续时间概率分布,即

状态的状态持续时间概率分布 $f_i(l)$ 为:

$$[0123] \quad f_i(l) = \alpha\beta (\alpha l)^{\beta-1} e^{-(\alpha l)^\beta};$$

[0124] 式中: α 、 β 分别为威布尔分布的尺度参数和形状参数;

[0125] 将状态监测值概率分布连续化。同样设定其服从威布尔分布,状态检测值概率分布函数 $\xi_i(\theta)$ 为:

$$[0126] \quad \xi_i(\theta) = P(x_t = \theta | q_t = x_t) = \alpha_i \beta_i (\alpha_i \theta)^{\beta_i-1} e^{-(\alpha_i \theta)^{\beta_i}};$$

[0127] 其中 α_i 、 β_i 为各状态阶段的威布尔分布的参数;故改进的HSMM模型可描述为 $\lambda = (\pi, f_i(l), \xi_i)$ 。

[0128] 步骤4,故障预测。

[0129] 假设的故障和非故障模型进行训练,即参数 λ_F 和 $\lambda_{\bar{F}}$ 。目标是评估,给定一个观察序列(错误序列) $O = [o_1, o_2, \dots, o_1]$ 是否为故障相关序列。首先计算分类模型的序列似然值,随后被分类为无故障或故障贝叶斯决策理论。

[0130] 步骤5,故障结果预判:

$$[0131] \quad \log[P(o|\lambda_F)] - \log[P(o|\lambda_{\bar{F}})] > \log \left[\frac{C_{\bar{F}\bar{F}} - C_{\bar{F}F}}{C_{F\bar{F}} - C_{FF}} \right] + \log \left[\frac{P(\bar{F})}{P(F)} \right];$$

[0132] 上面公式成立时,将一个序列标记成为故障相关事件序列,系统发出故障预测。其中 $C_{\bar{F}\bar{F}}$ 表示错误的将故障相关序列判断成为故障无关序列的代价, $P(F)$ 表示故障的概率, $\log[P(o|\lambda_F)]$ 表示对序列似然值取对数,这样可防止序列似然值太小而发生溢出问题。通过这样的方法,可以对每个序列进行判断,做出故障预测。

[0133] 以上所述仅为本发明的较佳实施例而已,并不用以限制本发明,凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明的保护范围之内。

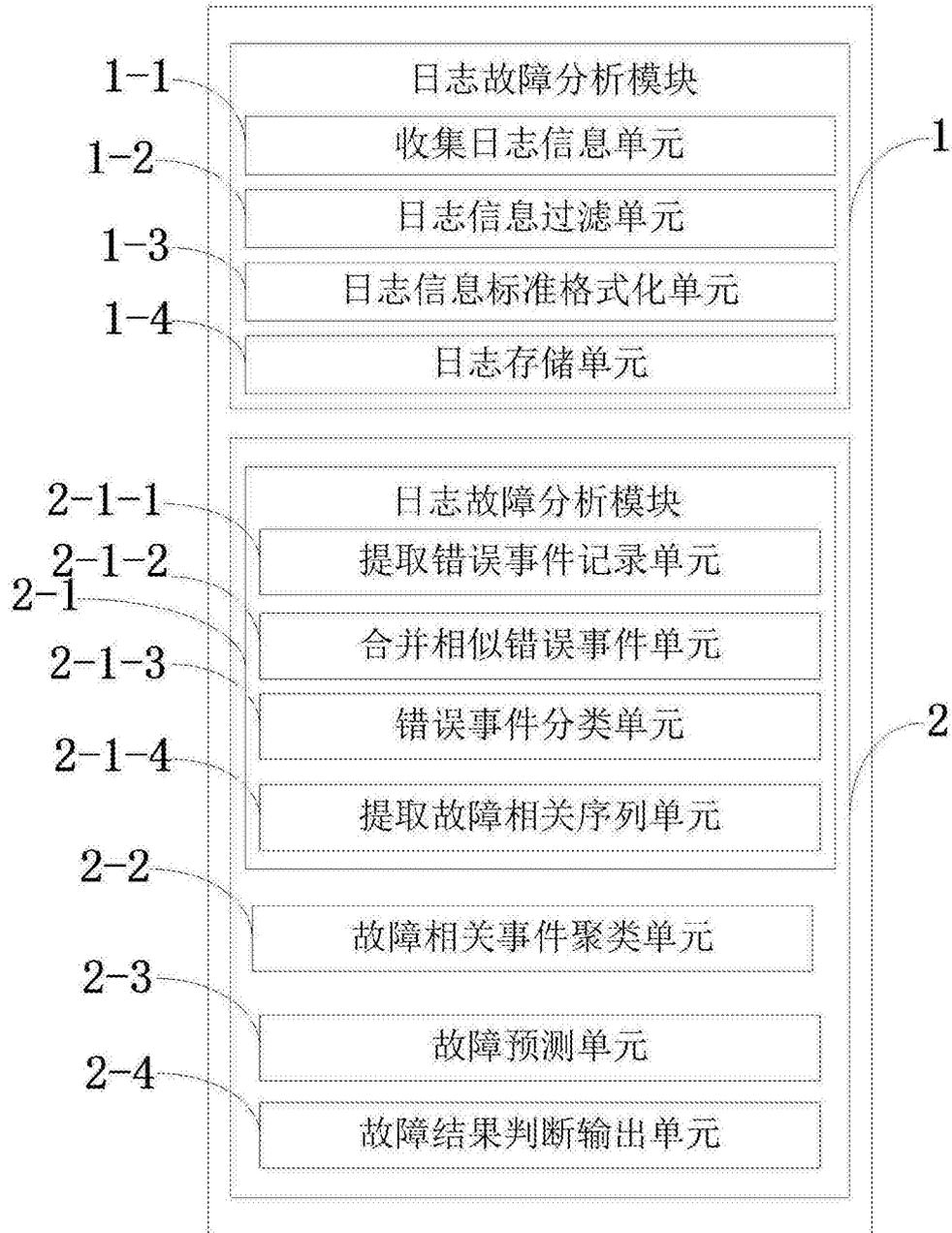


图1

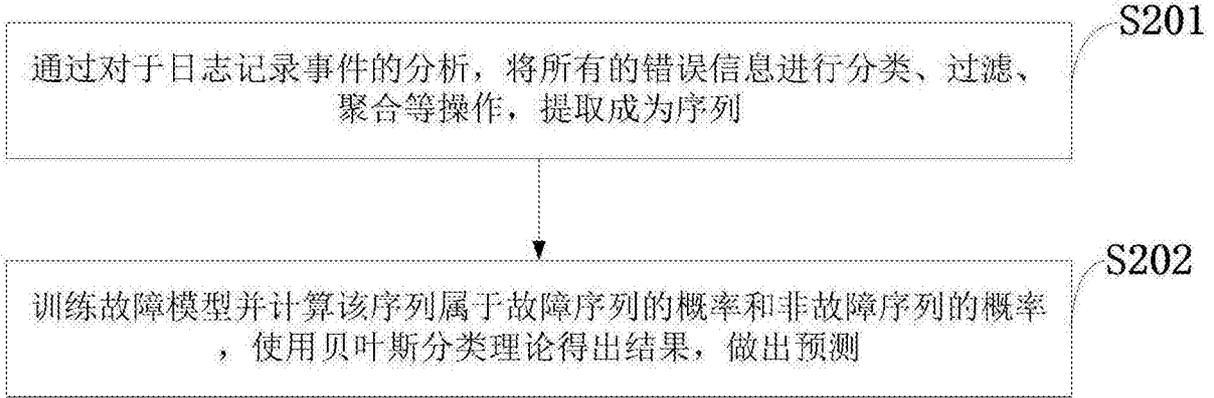


图2

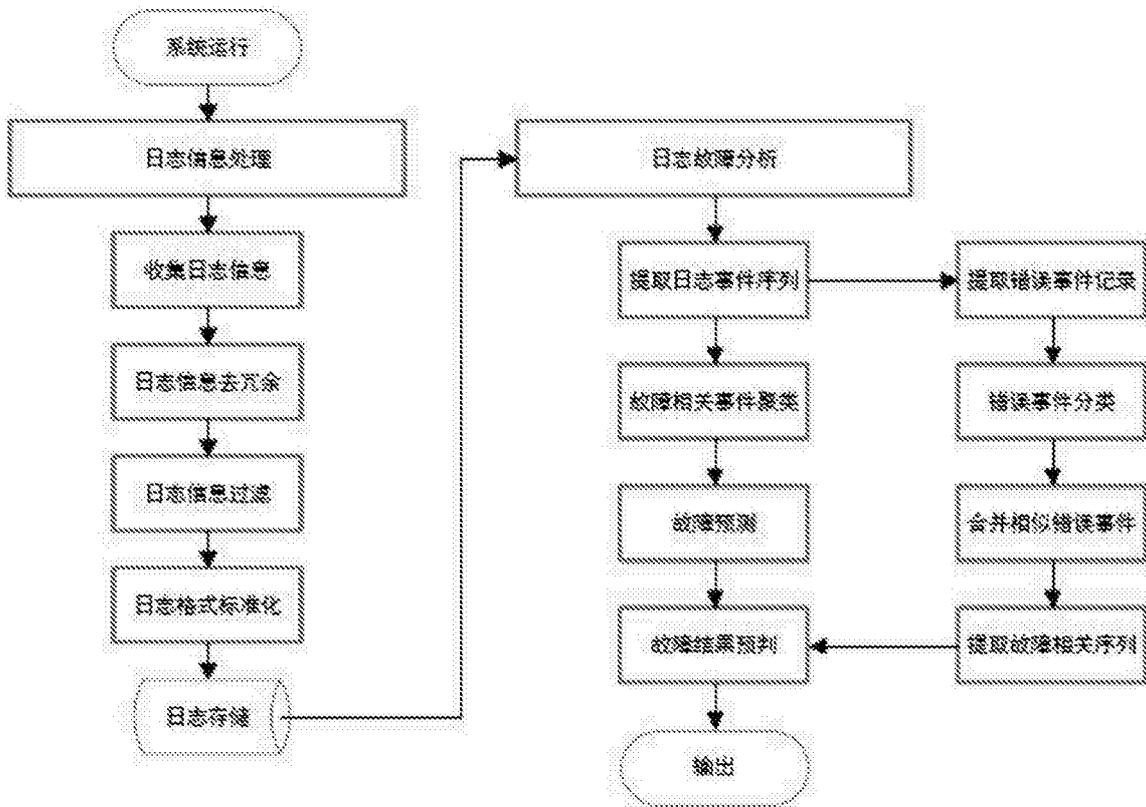


图3

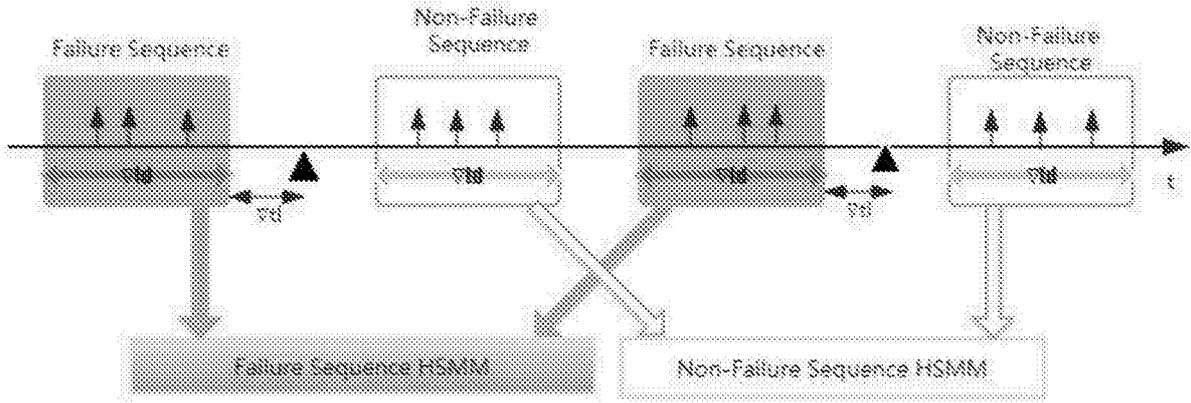


图4