



(12) 发明专利申请

(10) 申请公布号 CN 103632197 A

(43) 申请公布日 2014. 03. 12

(21) 申请号 201310535743. 4

(22) 申请日 2013. 11. 01

(71) 申请人 国家电网公司

地址 100031 北京市西城区西长安街 86 号

申请人 国网电力科学研究院

国网重庆市电力公司

(72) 发明人 方印 王俊梅 郭琳 史述红

高春成 于贺 王峰 李守保

王京 赵显

(74) 专利代理机构 北京中济纬天专利代理有限

公司 11429

代理人 张晓霞

(51) Int. Cl.

G06Q 10/00 (2012. 01)

G06Q 50/06 (2012. 01)

G06F 9/44 (2006. 01)

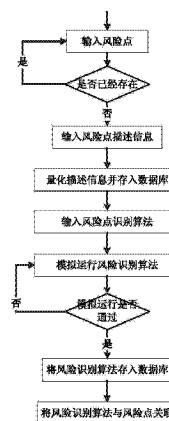
权利要求书2页 说明书7页 附图4页

(54) 发明名称

一种电力交易信息发布风险识别方法

(57) 摘要

本发明属于电力信息发布领域,涉及一种电力交易信息发布风险识别方法解决发布信息过程中的风险识别问题。风险识别系统使用下面几个步骤进行解决:(1) 风险点定义,将风险量化,将量化数据以属性的形式存入数据库风险点表中。(2) 风险识别算法设计,利用面向风险识别的程序语言,设计相应的风险点的识别算法。(3) 执行风险识别算法,在风险识别时机到来时,由系统自动调用风险识别算法执行。(4) 风险识别结果发布,根据第三步骤执行的结果,通过识别系统的结果发布模块对信息发布人员进行预警。



1. 一种电力交易信息发布风险识别方法,其特征在于,所述方法步骤如下:

步骤 1:风险点定义,以风险点的形式存储在数据库中,这样在执行风险检测算法的时候可以直接从数据库中取出风险点,根据风险点的属性执行识别算法;

步骤 (11) 确定要检测的风险,定义风险的名字即风险点,添加风险点到风险点库中,系统先判断是否存在重名的风险点,如果存在重名的风险点,要重新输入一个别的风险点名;

步骤 (12):如果不存在重名,系统将风险点加入风险点库中,并要求输入该风险点的描述信息,描述信息主要包括风险类型,风险级别,风险识别优先级,风险识别开始时间,风险识别激活间隔时间,涉及的数据库表、表中的字段、字段的期望值、风险影响指数、该字段的风险影响因子,风险报警级别,风险报警通知人;

步骤 (13):描述信息输入之后,系统会将描述信息量化,即转变成适合数据库存储的方式,存入数据库中;

步骤 (14):接着系统要求输入风险识别算法,算法的具体编写过程在步骤 2 中介绍,系统接收到风险识别算法后会进行一个模拟运行,这个运行主要是测试算法是否能够正确执行,是否能正确结束,算法在模拟运行的过程中不修改数据库中的任何信息,只有在算法被风险识别模块调用其运行才会影响数据库表中的信息;

步骤 (15):验证正确后,将算法存入数据库中,并与对应的风险点关联;

步骤 2:风险识别算法的定义,根据定义的风险点的特性,定义该风险点的识别算法;

步骤 (21):定义存储风险点量化信息的变量,从风险点库中取出该风险点的量化信息并为对应的变量赋值;

步骤 (22):从这些量化信息中解析出该风险点关联的数据库表及相关字段;

步骤 (23):进入一个循环,从数据库表中取出相关字段的值,与该字段在风险点库中存储的期望值进行比较或进行计算,根据结果计算得到该字段的风险影响指数,再乘以该字段对风险的影响因子,得到该字段的风险值,将值累加到风险预警级别,处理结束后取下一个字段执行同样的操作,直到处理完所有的字段;

步骤 (24):将得到的风险预警级别及每个字段的风险值存储到结果库表中,供以后对风险分析;

步骤 3:风险识别,在风险点识别时机到来的时候,调用并执行风险点识别算法,风险点识别算法执行结束后,调用风险执行结果发布模块,向有关人员发布风险预警;

步骤 (31):系统为每个添加进风险库中的风险点设立定时器,每当风险点的检测时机到来的时候,系统将该风险点加入风险识别队列,这是一个按风险点的风险识别优先级排序的一个优先级队列;

步骤 (32):风险点入队列后会检测风险识别模块是否休眠,如果识别模块正在休眠,激活风险识别模块,如果风险识别模块未休眠,则不进行任何操作;

步骤 (33):风险模块在未休眠状态时,不断的从风险识别队列中取出风险点然后执行,执行过程为,根据风险点 ID 取出风险识别算法,调用算法运行引擎,算法引擎加载风险识别算法并运行,运行结束算法会自动将运行结果信息加入风险识别结果库中,并退出算法运行引擎,将控制权交给风险识别模块;

步骤 (34):风险识别模块取出风险识别结果中风险预警级别,并与该风险点的报警级

别进行比较,如果风险预警级别交高,将风险预警信息发给风险报警通知人,如果风险预警级别较低,就由风险应对模块自动处理风险,处理过程根据风险识别结果库中的数据对需要发布的信息进行微调;

步骤(35):处理结束后风险识别模块从风险识别队列上取下一个需要检测的风险点,如果能够取到风险点,执行上述风险识别过程,如果未取到风险点,风险识别模块进入休眠状态。

一种电力交易信息发布风险识别方法

[0001]

技术领域

[0002] 本发明属于电力信息发布领域,涉及一种电力交易信息发布风险识别系统解决发布信息过程中的风险识别问题。

[0003]

背景技术

[0004] 由于电力的自然垄断属性,我国的电力交易注意由国家电网和南方电网对其负责区域内的电力进行组织和实施,因此电网担负着为社会提供安全可靠的电力的责任,这就要求电网能够合理安全的协调电力交易。在电力交易的过程中,电网要向电力客户提供一定的信息,使电力客户能够根据电网提供的信息进行决策。为了对电力客户提供高质量的,电网在发布信息的时候应该对将要发布的信息进行安全审核。

[0005] 电力交易机构必须向电力市场主体提供充足的信息,在向外公布信息的同时,要对公布的信息进行安全审核。国家电网信息发布中存在的风险主要由两种,第一种是信息合规风险,第二种是信息不安全扩散风险。信息合规风险,指的是根据国家有关监管部门的规定,必须在规定的时间向外公布足够的信息,如果电力交易机构没有按照规定,有可能带来合规风险。信息不安全扩散风险,在信息配置技术中介绍过,指的是向不同的市场主体提供不同的信息,如果向一些市场主体提供了他们不应该知道的信息,或者未向一些市场主体提供他们应该获得的信息,都有可能引发电力市场运行危机,带来信息不安全扩散风险。这两种信息发布风险,都会给电力市场带来不利的影响,都有可能阻碍清洁能源的消纳与健康发展。

[0006] 我国电力交易系统由三级交易机构组成,分别为国家电力交易机构、区域电力交易机构、省级电力交易机构。每个层次的电力交易机构的职能和面向的电力客户不同,因此向电力客户提供信息也不仅相同。但是国家电网为了维护其负责区域内的电力交易能够安全有效的进行,会对三级交易进行协调管理,自然包括其发布信息的管理以及发布信息风险识别。

[0007] 由于我国电力交易涉及的人员众多,需要发布的信息也复杂多样,因此在信息发布的风险识别,如果没有一个很好的解决办法,仅靠对发布信息的人工校验,会给电力交易带来重大的人力浪费,也会带来重大的安全隐患。

[0008] 为了有效的解决发布信息的风险识别问题,必须要建立一套自动的风险识别系统,这个系统必须能够准确的检测出发布信息中存在的风险;而且系统必须具有良好的扩展性,以应对新的风险;系统必须具有预警和评级机制,可以使风险检测人员能够及时的发现风险和了解风险的危害程度。

[0009] 对电力信息进行自动风险识别是一个全新的课题,没有成型的技术或者系统可以借鉴,以往对发布信息的风险识别都是通过信息发布人员与电力交易人员进行协调沟通进

行风险分析以及通过主管人员对发布信息进行确认,这样的检测方法是低效的,同时也较容易出现错误。

[0010]

发明内容

[0011] 信息发布系统的风险识别的难点在于存在的风险种类繁多,并且不同风险的检测时机不同,不同风险的检测涉及的人员关系不同,而且随着信息系统的发展发布信息中存在的风险也会相应的变化。这些问题为风险识别系统的设计提出了几个要求,那就是能够适应多种类的风险检测、能够适应风险种类的变化。

[0012] 针对上述问题,风险识别系统使用下面几个步骤进行解决:

(1) 风险点定义,将风险量化,将量化数据以属性的形式存入数据库风险点表中。

[0013] (2) 风险识别算法设计,利用面向风险识别的程序语言,设计相应的风险点的识别算法。

[0014] (3) 执行风险识别算法,在风险识别时机到来时,由系统自动调用风险识别算法执行。

[0015] (4) 风险识别结果发布,根据第三步骤执行的结果,通过识别系统的结果发布模块对信息发布人员进行预警。

[0016] 本发明针对上述现有技术中遇到的问题,开发了一套风险识别系统,即基于脚本语言的风险识别系统,风险识别系统识别的设计思路如下:

将现存的风险量化,以风险点的形式存储在数据库中的风险点表中,风险点表的属性即为风险各种量化的参数。为每个风险点设计风险识别算法,算法的设计采用面向风险识别的程序设计语言设计,这种语言为风险识别算法的设计提供了更方便的内置参数类型,面向风险识别的程序设计语言也是本发明的一个重要部分。风险检测算法和风险点定义完毕之后,根据风险点的属性获得风险检测的时间点,在时间点到来的时候,风险识别系统自动调用并执行风险检测算法。风险检测算法是在专门为算法语言设计的解释器上运行的,运行结果存储到数据库中相应的表格内,并通知识别系统进行检测结果的发布。运行检测算法的解释器也是为本发明专门设计的,将算法解释成 java 语言在系统的 java 虚拟机上运行。

[0017] 基于脚本的风险识别系统的框架结构图如附图 1 所示,其中各模块的作用及它们之间的相互关系如下:

(1) 风险识别系统总体结构

风险识别系统包含四个子模块。第一个为风险点管理模块,这个模块实现风险点的录入、修改和删除功能。第二个为风险检测模块,主要完成从数据库中取出对应风险点的风险识别算法,并调用运行引擎执行算法。第三个为风险检测算法的运行引擎,主要为风险检测算法的运行提供运行环境。第四个为风险预警和风险应对模块,主要完成对风险识别算法执行结果的发布与通告。

[0018] (2) 风险点管理模块

风险管理模块主要实现风险点数字化,并提供风险点的定义、修改和删除功能,以及风险识别算法的定义。由于每个风险开始并不是数字化的,而且风险点的数字化是一个很复

杂的过程,如果让系统的使用者将风险点数字化后再录入风险数据库时不现实的,因此提供这个风险点管理模块。下面分别描述风险点的定义、修改和删除流程,及风险识别算法的定义。

[0019] 1) 风险点的定义。

[0020] 在一种新的风险点需要定义时,系统使用者访问该模块,定义风险点名,模块自动检测是否与风险库中的风险点是否存在重名,如果不重名,输入风险点的各种描述信息。

[0021] 管理模块根据输入的信息,将风险点数字化,并将数字化后的信息录入数据库。

[0022] 2) 风险点的修改。

[0023] 要修改一个风险点时,首先检测该风险点是否被风险点检测模块调用,如果被调用则不允许修改;如果不存在调用则开放修改,可以修改风险点名,以及各种属性;修改结束后该模块将风险点重新数字化后修改风险点在数据库中的改变的属性。

[0024] 3) 风险点的删除。

[0025] 要删除一个风险点时,首先检测该风险点是否被风险点检测模块调用,如果被调用则不允许删除;如果不存在调用则开放删除;删除后由该模块从数据库中将该风险点删除。

[0026] 4) 风险识别算法定义

首先确定一个检测算法的功能,明确算法要检测的风险点;将相关的风险点关联到算法;编写算法的具体实现;对算法模拟运行以检测算法是否能正确的执行;算法模拟运行正确后将算法以及关联的风险点存入数据库。

[0027] (5) 风险检测算法运行引擎

这个模块是面向风险识别设计语言的解释执行引擎。在风险识别过程中调用风险识别算法,自动加载算法需要的数据,并解释执行,执行结束,将运行得到的风险数据存入风险检测结果库内,并通知风险检测模块,对检测结果进行警告或应对。

[0028] (6) 风险的数据库存储

风险数据库主要存储将风险识别系统需要的各种信息,主要包括下面几个数据库表格:

1) 风险点描述信息表。这个表主要是存储风险点定义时录入的描述信息,这些描述信息可以供风险点管理模块将风险点数字化提供依据。

[0029] 2) 风险点数字化表。这个表主要是将风险点数字化,为风险检测算法的计算提供依据。

[0030] 3) 风险检测算法表。这个表中主要存放实现各种风险检测算法。

[0031] 4) 风险检测结果表。这个表主要存放风险检测算法执行之后的结果信息。

[0032] 5) 风险检测算法与风险点数字化表关联表。这个表主要存放风险检测算法与风险点之间的关联关系,以实现检测算法与风险点的多对多关系。

[0033] (7) 风险警告和应对

通过研究经典的人工神经网络识别风险识别技术,实现面向长周期大范围清洁能源消纳的市场发布信息风险识别功能,并以颜色分级等方式进行风险的预警和监控。

[0034] 风险识别系统识别风险的步骤如下:

(1) 风险点定义:将风险量化,以风险点的形式存储在数据库中,这样在执行风险检测

算法的时候可以直接从数据库中取出风险点,根据风险点的属性执行识别算法。

[0035] (2) 风险识别算法定义:根据定义的风险点的特性,定义该风险点的识别算法。

[0036] (3) 风险识别算法执行:在风险点识别时机到来的时候,调用并执行风险点识别算法。

[0037] (4) 风险执行结果发布:风险点识别算法执行结束后,调用风险执行结果发布模块,向有关人员发布风险预警。

[0038] 本发明的具体方法步骤如下:

步骤 1:风险点定义,以风险点的形式存储在数据库中,这样在执行风险检测算法的时候可以直接从数据库中取出风险点,根据风险点的属性执行识别算法。

步骤 (11) 确定要检测的风险,定义风险的名字即风险点,添加风险点到风险点库中,系统先判断是否存在重名的风险点,如果存在重名的风险点,要重新输入一个别的风险点名。

[0039] 步骤 (12):如果不存在重名,系统将风险点加入风险点库中,并要求输入该风险点的描述信息,描述信息主要包括风险类型,风险级别,风险识别优先级,风险识别开始时间,风险识别激活间隔时间,涉及的数据库表、表中的字段、字段的期望值、风险影响指数、该字段的风险影响因子,风险报警级别,风险报警通知人。

[0040] 步骤 (13):描述信息输入之后,系统会将描述信息量化,即转变成适合数据库存储的方式,存入数据库中。

[0041] 步骤 (14):接着系统要求输入风险识别算法,算法的具体编写过程在步骤 2 中介绍,系统接收到风险识别算法后会进行一个模拟运行,这个运行主要是测试算法是否能够正确执行,是否能正确结束,算法在模拟运行的过程中不修改数据库中的任何信息,只有在算法被风险识别模块调用其运行才会影响数据库表中的信息。

[0042] 步骤 (15):验证正确后,将算法存入数据库中,并与对应的风险点关联。

[0043] 步骤 2:风险识别算法的定义,根据定义的风险点的特性,定义该风险点的识别算法。

[0044] 步骤 (21):定义存储风险点量化信息的变量,从风险点库中取出该风险点的量化信息并为对应的变量赋值。

[0045] 步骤 (22):从这些量化信息中解析出该风险点关联的数据库表及相关字段。

[0046] 步骤 (23):进入一个循环,从数据库表中取出相关字段的值,与该字段在风险点库中存储的期望值进行比较或进行计算,根据结果计算得到该字段的风险影响指数,再乘以该字段对风险的影响因子,得到该字段的风险值,将值累加到风险预警级别,处理结束后取下一个字段执行同样的操作,直到处理完所有的字段。

[0047] 步骤 (24):将得到的风险预警级别及每个字段的风险值存储到结果库表中,供以后对风险分析。

[0048] 步骤 3:风险识别,在风险点识别时机到来的时候,调用并执行风险点识别算法,风险点识别算法执行结束后,调用风险执行结果发布模块,向有关人员发布风险预警。

[0049] 步骤 (31):系统为每个添加进风险库中的风险点设立定时器,每当风险点的检测时机到来的时候,系统将该风险点加入风险识别队列,这是一个按风险点的风险识别优先级排序的一个优先级队列。

[0050] 步骤 (32) :风险点入队列后会检测风险识别模块是否休眠,如果识别模块正在休眠,激活风险识别模块,如果风险识别模块未休眠,则不进行任何操作。

[0051] 步骤 (33) :风险模块在未休眠状态时,不断的从风险识别队列中取出风险点然后执行,执行过程为,根据风险点 ID 取出风险识别算法,调用算法运行引擎,算法引擎加载风险识别算法并运行,运行结束算法会自动将运行结果信息加入风险识别结果库中,并退出算法运行引擎,将控制权交给风险识别模块。

[0052] 步骤 (34) :风险识别模块取出风险识别结果中风险预警级别,并与该风险点的报警级别进行比较,如果风险预警级别交高,将风险预警信息发给风险报警通知人,如果风险预警级别较低,就由风险应对模块自动处理风险,处理过程根据风险识别结果库中的数据对需要发布的信息进行微调。

[0053] 步骤 (35) :处理结束后风险识别模块从风险识别队列上取下一个需要检测的风险点,如果能够取到风险点,执行上述风险识别过程,如果未取到风险点,风险识别模块进入休眠状态。

[0054] 本发明的有益效果在于 :

电力交易信息发布风险识别方法,是在对现有的信息发布风险人工排查的分析,并结合信息技术提出的一种风险自动识别方法。使用这种方法能够极大的减轻人工排查风险的负担,又比人工风险排查具有更高的准确率。随着电力信息发布业务的增加,相应的发布风险也随着增多,风险点的增多不仅是数量上,而且种类上也会不断的增加,该方法在发明的时候就考虑到对风险点的扩展,能够最大限度的适应风险点数量和种类的增加,可以在不修改系统的前提下扩充风险点库和风险检测算法。

[0055] 附图说明 :

图 1 为本软件系统的模块结构图。

[0056] 图 2 为风险点定义流程图。

[0057] 图 3 为风险识别算法编写流程图。

[0058] 图 4 为风险识别过程。

[0059]

具体实施方式

[0060] 作为一个例子,我们以月报的及时性风险为例,来详细介绍利用风险识别系统进行对风险点量化和风险识别算法的编写。

[0061] 1. 首先确定风险点的名称为月报及时性风险,然后确定风险的类别为及时性风险,再加入风险的描述信息,形成下面这样一个风险描述信息表 :

描述	信息
风险点名称	月报及时性风险
风险类别	及时性风险
风险识别优先级	中
风险识别开始时间	每月 25 日左右
风险识别间隔时间	一个月
涉及的数据库表、字段、期望值、查询条件、查询字段、风险影响指数、风险影响因子	报表表、报表提交时间、当月 27 日之前、报表类别、月报、1、1
报警级别	0.8
报警通知人	系统管理员
通知方式	系统弹窗

根据上表描述信息,系统自动将描述信息良好后存储在数据库各表中,如下所示 :

(1) 风险点表 (risk)

字段名	值
ID	000001
Type	0001(对应风险类别表中 ID 字段 0001)
Priority	3(对应中度报警级别)
Start_time	0000:00:27(YYYY:MM:DD)
Interval_time	0000:01:00(YYYY:MM:DD)
Alarm_level	0.8
Alarm_staff	0001(对应系统人员表中 ID 字段 0001)
Alarm_type	0001(对应报警级别表中 ID 字段 0001)

(2) 风险识别涉及表字段等信息表 (involve_relation)

字段名	值
ID	0000001
Risk_id	000001
Table	Report(报表表)
Field	Sumit_time(提交时间)
Expect	0000:00:27(YYYY:MM:DD)
Check_field	Type
Check_field_value	Month
Risk_influence_level	1
Risk_influence_rate	1

将风险描述信息量化并存入数据库后,就可以风险识别算法的编写了。

[0062] 2. 下面这个风险识别算法是利用面向风险识别的脚本语言编写的针对月报及时性风险的算法,算法中使用的一些函数,是为了风险识别算法编写的简便性提供的库函数。

[0063] riskIdentified() {

 // 定义需要用到的变量

 ID risk_id; // 风险点 ID

 String check_table; // 需要检查的表

 String field; // 需要查询的字段

 String check_field; // 查询条件字段

 String check_field_value; // 查询条件对应的值

 Time sumit_time, expect_time, cureent_time; // 提交时间、期望时间及当前时间

前时间

 Float risk_influence_level ; // 风险影响指数

 Float risk_influence_rate; // 风险影响因子

 Float alarm = 0; // 预警级别并初始化为 0

 // 为一些变量赋值

 risk_id = getID();

 cureent_time = getCurrentTime();

 expect_time = getTimeByCurrent("involve_relation", "risk_id", risk_id, "expect");

 risk_influence_level = getFloat("involve_relation", "risk_id", risk_id, "Risk_influence_level");

 risk_influence_rate = getFloat("involve_relation", "risk_id", risk_

```
id, "Risk_influence_rate");
    check_table = getString("involve_relation", "risk_id", risk_id,
"Table");
    field = getString("involve_relation", "risk_id", risk_id, "Field");
    check_field = getString("involve_relation", "risk_id", risk_id,
"Check_field");
    check_field_value = getString("involve_relation", "risk_id", risk_
id, "Check_field_value");
    // 计算预警级别
    sumit_time = getTimeByCurrent(check_table, check_field, check_field_
value, field);
    if(sumit_time == null || sumit_time > expect_time){
        alarm = 1 * risk_influence_level * risk_influence_rate;
    }
    // 写结果表
    String insertSql = "insert into risk_result(risk_id, time, alarm)
values(" + risk_id, current_time, alarm + ")";
    executeInsert(insertSql);
}
```

算法编写完成之后存入算法表总,并将该算法对应的 risk_id 设置为该风险点的 id。

[0064] 下面对上述算法使用的库函数做进一步的说明：

getCurrentTime() 获取系统当前时间,返回一个 YYYY:MM:DD 格式的时间。

[0065] getTimeByCurrent(String table, String field, String field_value, String expect_field) 根据数据库表名 table, 查询条件字段 field, 查询条件字段值 field_value, 获取字段 expect_field 中的 Time 类型的值。

[0066] getFloat() 与 getTimeByCurrent 的参数相同,返回查询字段中的浮点数值。

[0067] getString() 与 getTimeByCurrent 的参数相同,返回查询字段中的字符串类型的值。

[0068] executeInsert(insertSql), 整型插入数据型 sql 语句 insertSql。

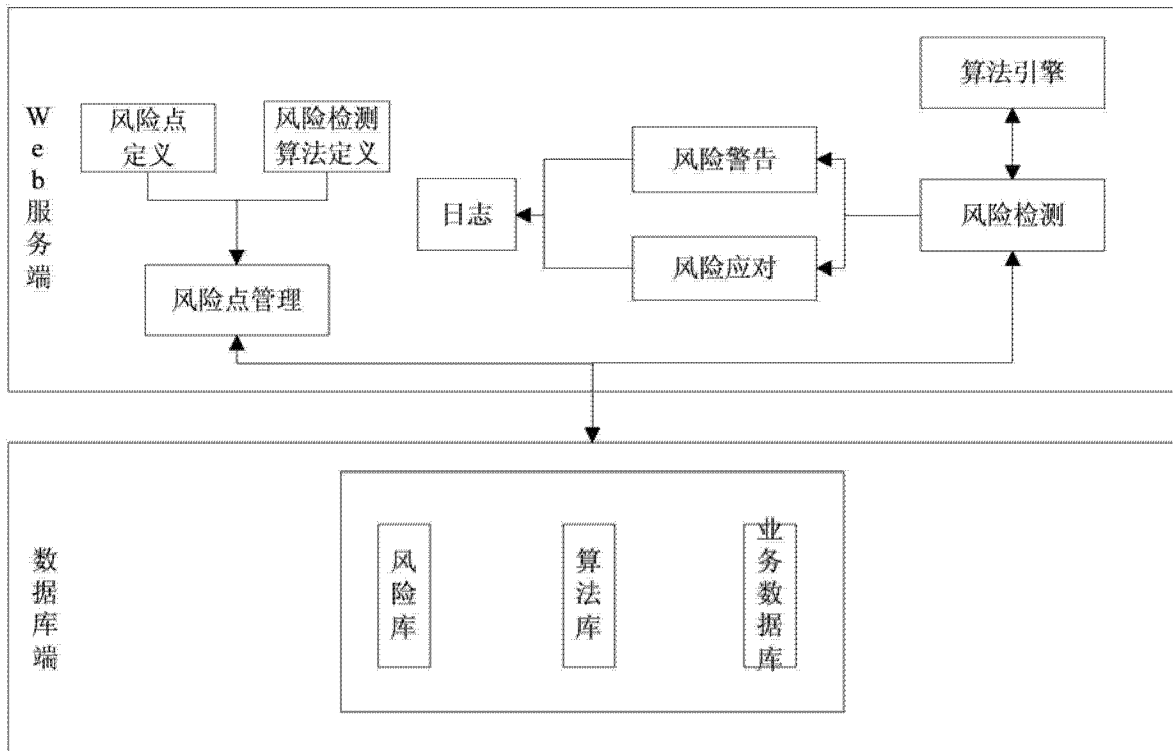


图 1

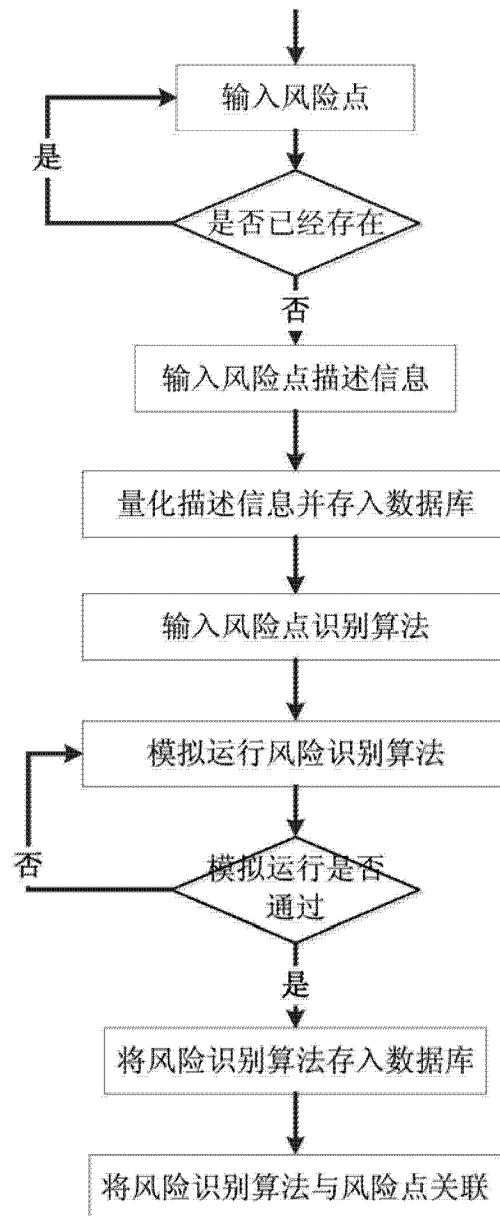


图 2

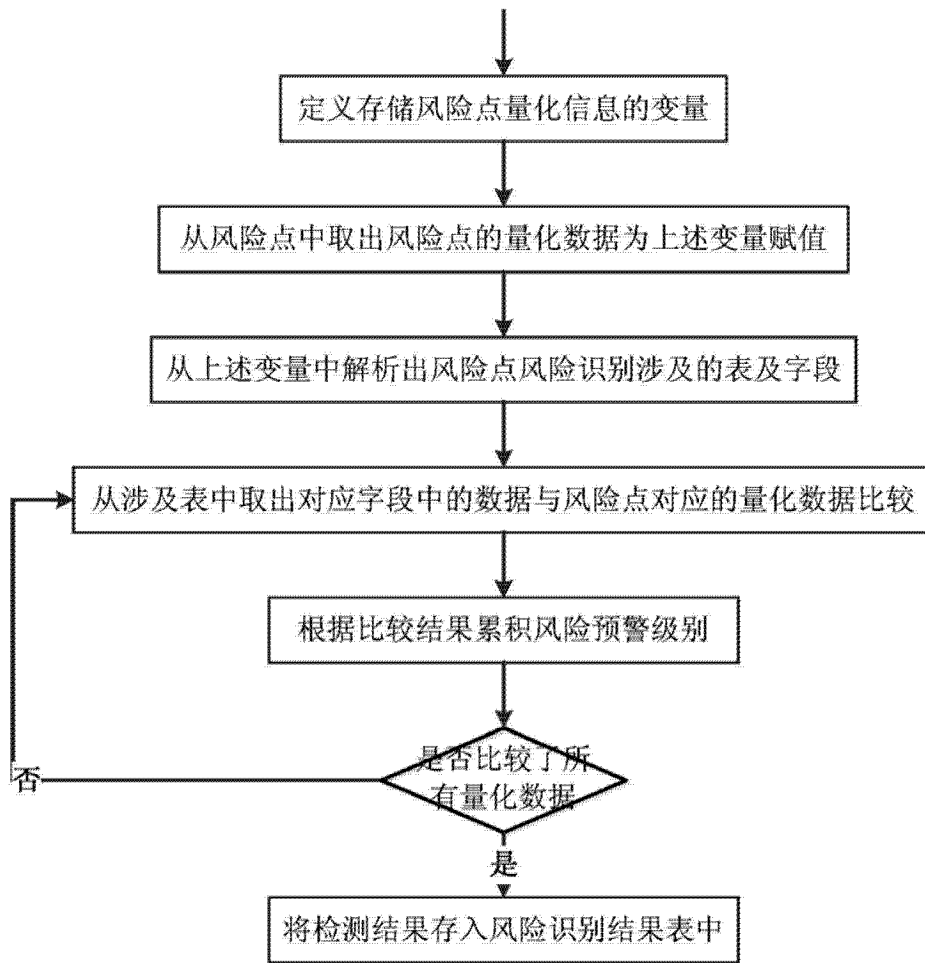


图 3

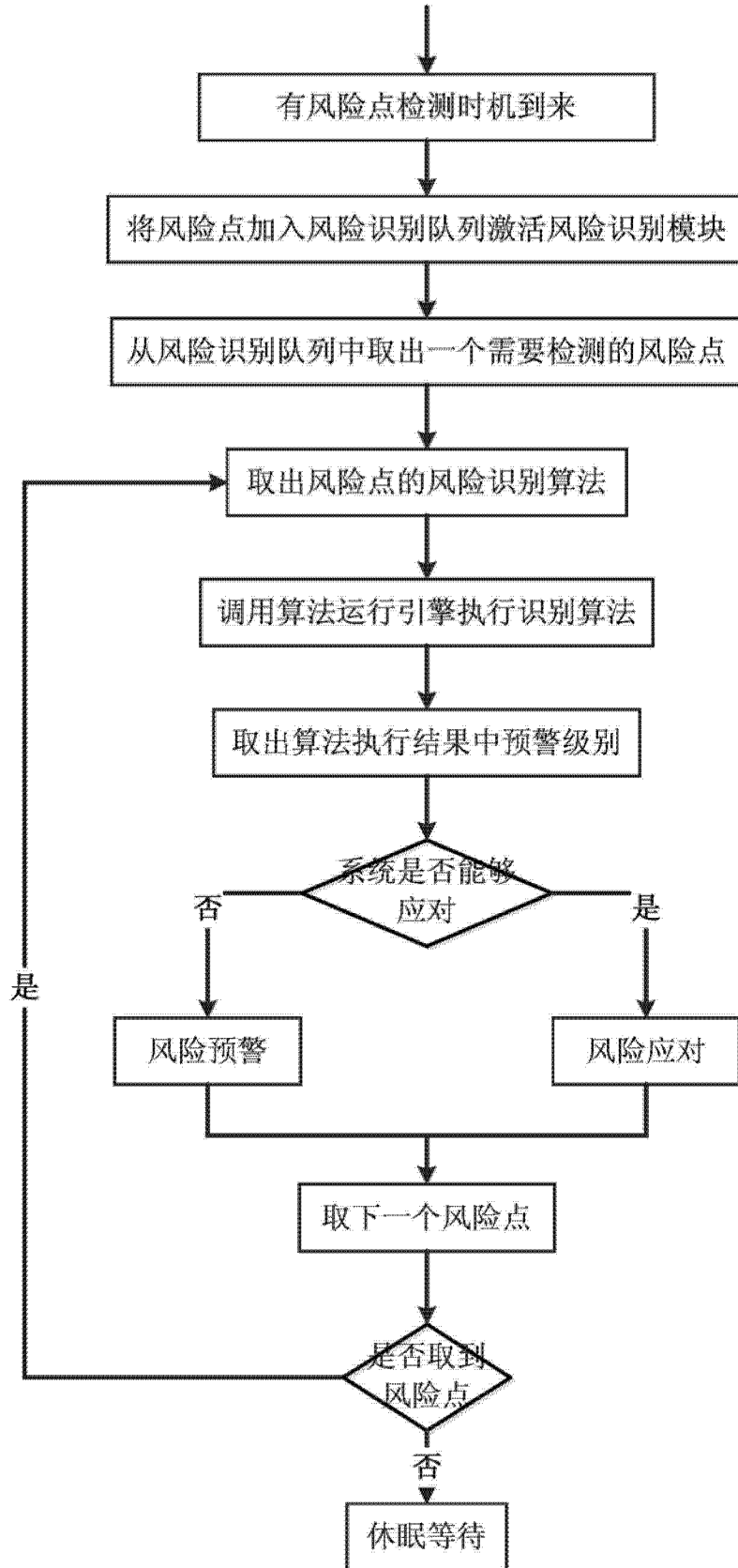


图 4