# 基于关系驱动的事件识别软件V1.0

# 说明书

**西安电子科技大学**

## 软件描述

随着信息技术的快速发展，具备自主性特征已经成为诸如军事系统、航天系统和

移动机器等软件系统发展的重要趋势，美国国防部于 2012 年在关于自主性的报告 （DSB/2012）中提出了面向自主系统的参考框架（autonomous system reference

framework，ASRF），为这类系统的理论、设计、实施与操作之间的相互联系提供了一定的依据。自主系统是指能够使用机器学习和人工智能等技术进行自我调控，而不 需或较少地需要人工干预的一类系统，该类系统能够完成自我管理和自我引导。自主系统一般通过感知、分析、决策和执行四个过程实现自我调整[3]。具体地，自主系统实时监测系统内外部资源以及运行环境的变化，识别由这些变化触发的异常事件，并产生和执行调控策略。分析过程在自主调整过程中处于承上启下的位置，其识别出的事件是后续决策与执行过程的依据。而自主系统通常是任务攸关型系统，如果在系统运行时识别结果存在错误或者延迟，则将导致逾期或错误等不可靠的调整，使得系统行为退化，甚至将系统引入更错误的状态中。系统工程师很难在软件设计与开发阶段对动态自主系统进行严格且完 整的测与验证，以提前保障自主系统的可靠性需求，这其中自然包括自主系统分析 过程的测试。分析过程通常发生在软件运行阶段，实时产生的识别结果是无法在软件运行前经过严格的测试与验证的，因此采用一种既准确又可靠的分析方法去识别事件显得尤为重要。基于关系驱动的事件识别软件可以既准确又可靠的识别事件。

## 软件功能

本软件提升自主系统软件事件识别的准确性，保障调整过程中分析环节的可性。主要包括：

* 潜在的事件关系挖掘与建模功能：通过模糊故障树建立事件关系模型，便于利用事件关系有效识别事件。
* 高准确事件识别功能：使用贝叶斯网络的事件识别方法，利用转换规则将模糊故障树转换为贝叶斯网络，以事件间的逻辑关系为依据，通过双向推理综合分析事件，提升事件识别的准确率。

## 3.软件的设计与实现

### 3.1 软件结构设计

（1）事件关系挖掘与建模方法

首先，研究事件关系挖掘方法，通过日志挖掘得到系统运行时事件与事件之间的

因果关系。然后，将系统运行时事件关系采用模糊故障树进行建模，支持向推理能力更强的贝叶斯网络的转换。

（2）基于贝叶斯网络的事件识别方法

首先，设计模糊故障树与贝叶斯网络之间的转换规则，将模糊故障树转换为贝叶

斯网络。然后，通过贝叶斯网络反向推理得到影响顶事件最大的基本事件，进行重点监控。最后，通过贝叶斯网路正向推理准确识别事件。

综上所述，研究内容一事件关系挖掘与模型建立，及研究内容二中贝叶斯网络模

型建立，均为系统实时事件识别之前的模型建立，指导研究内容二中反向推理与正向推理实现实时的事件识别。

### 3.2 软件流程描述

本软件的使用可分为两种组件开发流程来介绍：潜在的事件关系挖掘与建模功能与高准确事件识别功能：

**潜在的事件关系挖掘与建模功能：**将系统运行时事件关系采用模糊故障树进行建模，支持向推理能力更强的贝叶斯网络的转换。故障树（FT，Fault Tree）采用逻辑门连接上层与下层事件之间的因果关系，能够直观地表达本文抽取出的事件因果关系。在故障树模型中，故障树分析方法是一种通过演绎方式分析系统故障的方法。此方法包括建模和推理两个过程，其中建模过程将一个可能的故障作为顶事件，寻找所有可能导致顶事件发生的直接原因和间接原因，并分别作为中间事件和基本事件，用树状图中的各种逻辑门符号表达这些事件之间的逻辑关系。而推理过程为自底向上的正向推理，从已发生的基本事件向上推理得到可能发生的中间事件和顶事件。本软件将采用模糊故障树去建模事件间关系，实现不确定性推理。贝叶斯网络（BN，Bayesian Network）又称信念网络、因果网络或概率网络，由美国加州大学的 Pearl 于 1988 年首次提出。贝叶斯网络在具有坚固数学基础的同时又具有清晰的语义，它基于概率推理和图形理论实现智能诊断与预测，尤其在解决具有一定联系的不确定复杂故障上占据优势。贝叶斯网络具有极强的建模与推理能力，在推理时能够结合先验知识与当前实际监测值，已经在医疗诊断、机器故障诊断和计算机视觉等领域发挥重要作用。软件模型转换方法如下：当故障树越来越复杂时，用故障树分析就变得有局限性，并且推理能力也不足，只能逐层向上推理。利用贝叶斯网络可以提升故障树分析能力，因此软件通将故障树转换为贝叶斯网络进行故障分析。故障树模型向贝叶斯网络模型的转

换主要包括网络结构图转换和参数转换，其中网络结构图转换是指故障树的树结构转换为贝叶斯网络的图结构，参数转换是指故障树的逻辑门转换为贝叶斯网络的条件概率，以及故障树中基本事件的失效分布转换为贝叶斯网络中父节点的先验概率。

**高准确事件识别功能：**为能够基于事件因果关系识别事件，需清晰直观地建模 挖掘得到的因果关系。事件因果关系一般为单向关系，由底层基本事件指向高层复杂事件，适于采用树状图进行表示。本软件采用模糊故障树建模事件关系，首先将事件因果关系建模成模糊故障树中的树状图结构，然后计算模糊故障树中基本事件发生概率及模糊数，最后计算模糊故障树中的各个模糊算子。模糊故障树模型建立完成之后，本文将其交由领域专家进行审核、修改与完善，便于本文有效利用这些关系识别事件。本软件获取并建模 挖掘得到的各项事件因果关系，在这些因果关系中，结果事件中的高层复杂事件可作为树状图的最顶端事件，而事件因果关系将以父子层次结构进行组织与枚举。在得到模糊故障树中的树状图结构之后，软件统计这些树状图中的基本事件的历 史发生概率，并映射到三角模糊数上。具体地，本文首先通过日志统计的方式获取基本事件发生频率，以频率代替概率的方式得到基本事件发生概率。然后采用计算隶属度的方式，将基本事件发生概率映射到模糊数上。将基本事件发生概率映射到模糊数后，本软件计算模糊故障树中的所有模糊算子。本软件设计了以下处理措施去除多个故障树之间的关联关系。模糊故障树关联的情况可分为两种：一是不同模糊故障树中的顶事件之间存在因果关系，此时可将它们合并成为一棵模糊故障树；二是一棵模糊故障树中的基本事件连接至另一棵模糊故障树的中间事件，此时可在这两棵模糊故障树中分别建立这个基本事件节点，断开这种子树之间相连的边，两棵模糊故障树中中间事件互连的解决方法亦如此。通过上述处理，最终形成独立的多棵模糊故障树，每棵模糊故障树单独计算其模糊逻辑门算子，并且单独转换为贝叶斯网络，实现多棵模糊故障树分别独立与快速向上推理，降低事件推理过程的复杂性。软件使用贝叶斯网络反向推理获得对顶事件节点影响程度较高的基本事件节点进行重点监控，然后在系统实际运行时采用贝叶斯网络正向推理识别可能发生的异常事件，保障事件识别的及时性与准确性。

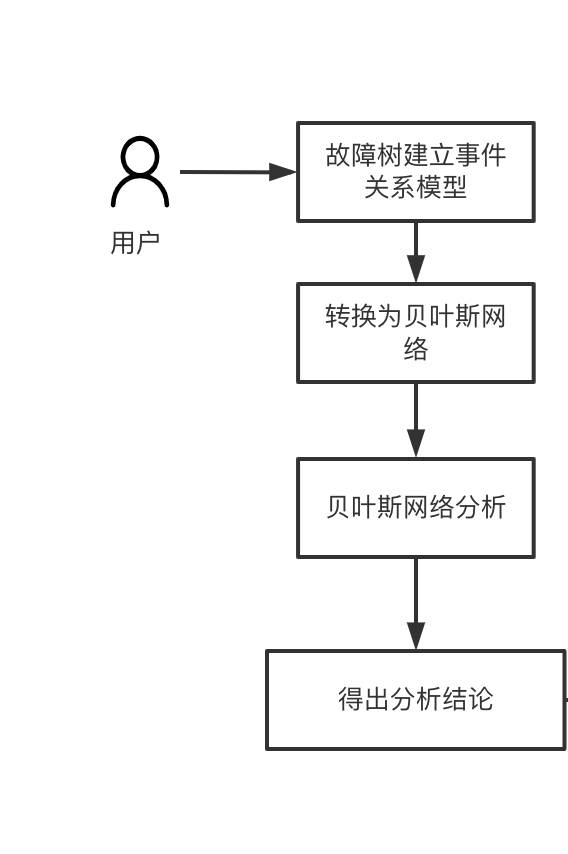


图1 潜在的事件关系挖掘与建模功能流程图

潜在的事件关系挖掘与建模功能流程如图1所示。用户在软件中使用故障树并进一步转换贝叶斯网络进行分析，得出潜在的事件关系模型。

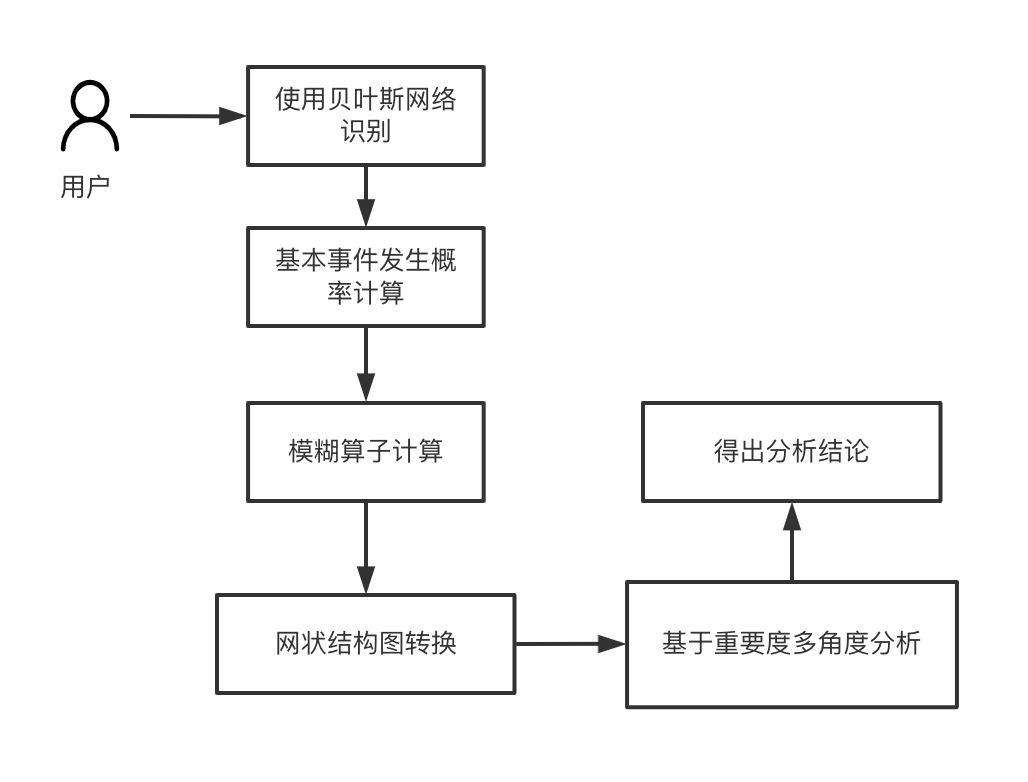


图2 高准确事件识别功能流程图

高准确事件识别功能流程如图2所示。用户在软件使用贝叶斯网络识别，并计算模糊因子，通过网状结构图和基于重要度多角度分析，最后实现事件的高准确识别。

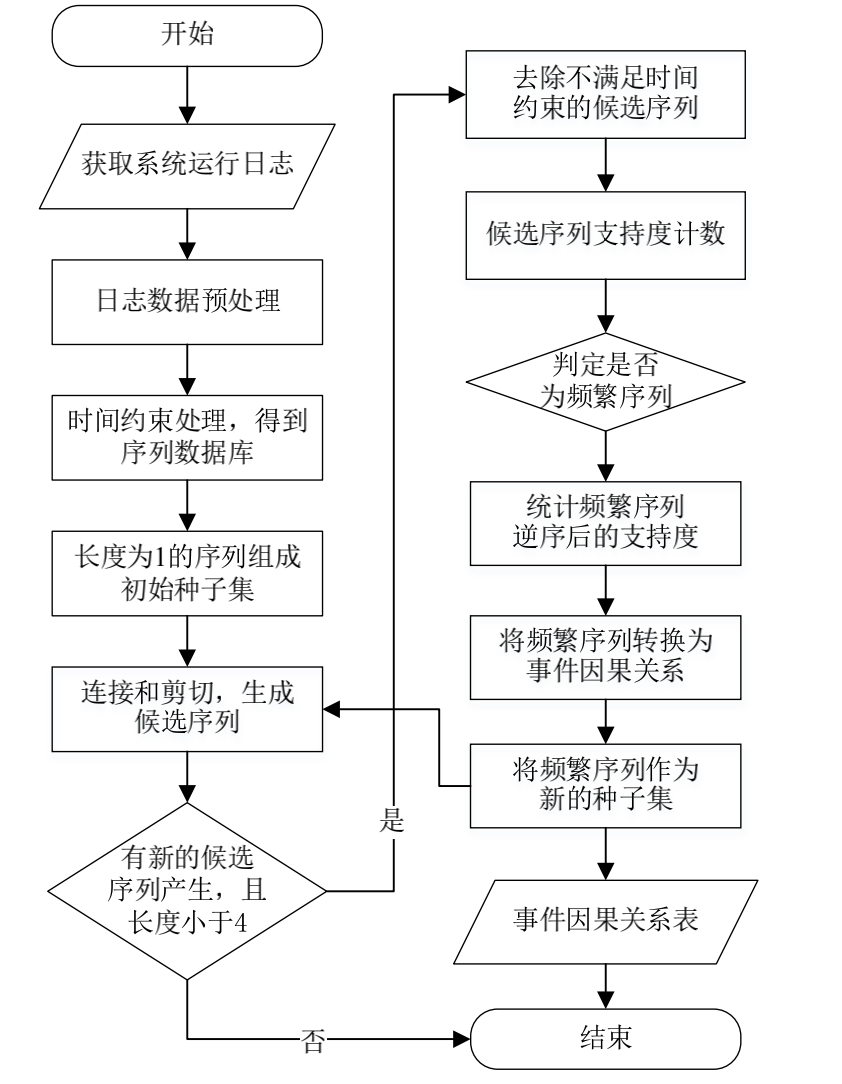


图3 事件关系挖掘方法流程图

### 3.3 典型应用

本软件为用户提供了基于关系驱动的事件识别平台软件，可以广泛的应用于基于于物联网、工业和一些航天航空等特定领域的开发中。

现以大型软件系统不稳定日志为例，介绍本软件的应用场景。

首先采用级联聚类将日志序列分组为类簇，使用贝叶斯网络识别，对类簇与 KPIs

（Key Performance Indicators，系统关键性能指标）进行关联性分析，识别大型在线服务系统中影响 KPIs 的问题，最后得出对事件的识别结论。

## 4.运行安装说明

由于本软件是以IDEA插件形式开发的，因此本软件的安装要依赖于IDEA软件的安装。

用户安装完成IDEA之后，才能安装本软件。IDEA插件安装的方式有许多中，本说明书就其中一种进行说明：

1. 准备好需要安装的本软件的插件文件。
2. 找到IDEA软件的安装目录，插件文件复制到自己的文件夹下。
3. 重新启动软件。

## 5.基于关系驱动的事件识别软件使用说明

### 5.1数据可视化展示平台软件主界面

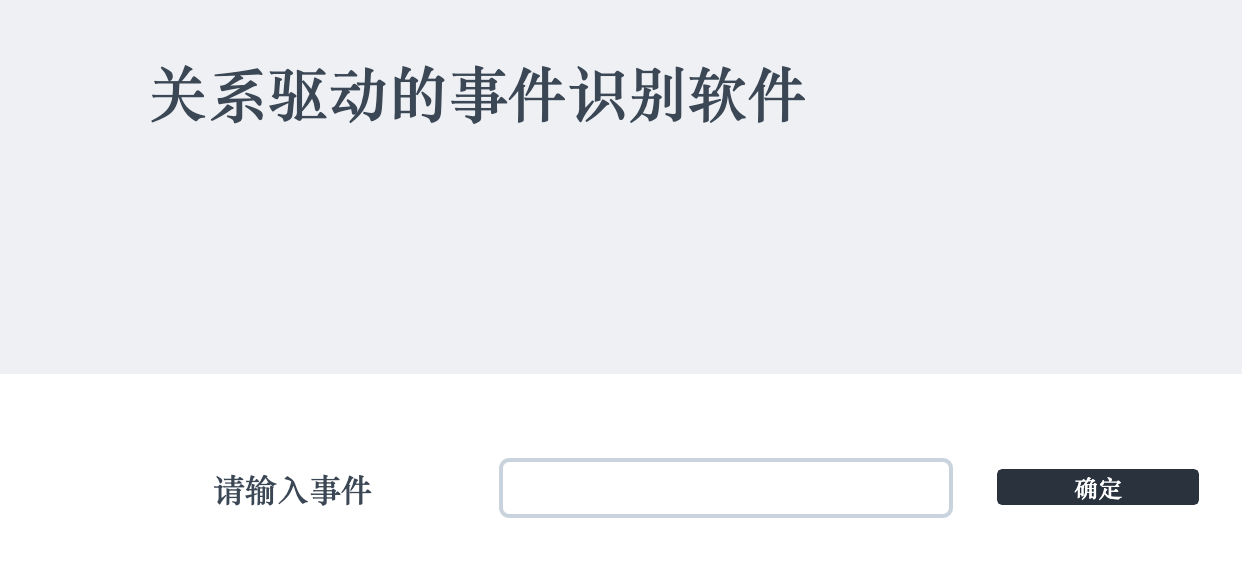


图4 系统主页面图

在系统的主页面输入事件，通过restful风格的API接口查询，将查询的事件以hash的形式发送请求返回到后端处理，获得查询结果，得到以json格式的编码信息后，对其进行反序列化，进行解析，最终将数据存到数据库中并返回给用户。

### 5.2 平台功能使用

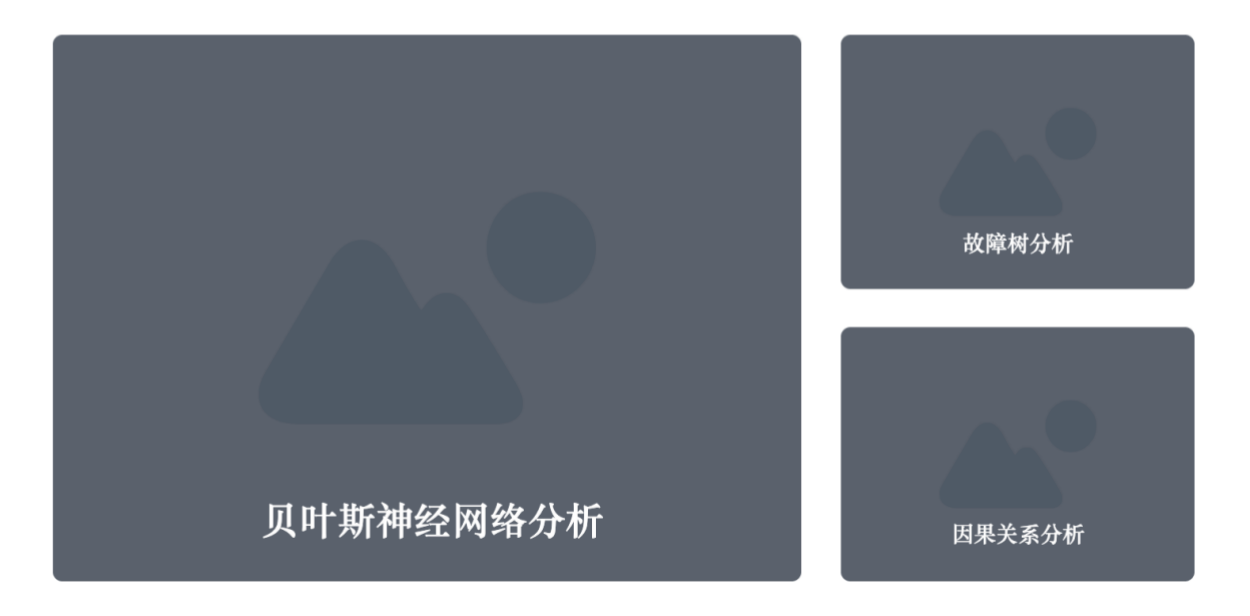


图5 选择分析页面

分析可选模糊树分析，也可以进一步转换成推理能力更强的贝叶斯神经网络分析。点击模糊数分析，通过序列化数据，程序开始分析，最后可得分析结果。

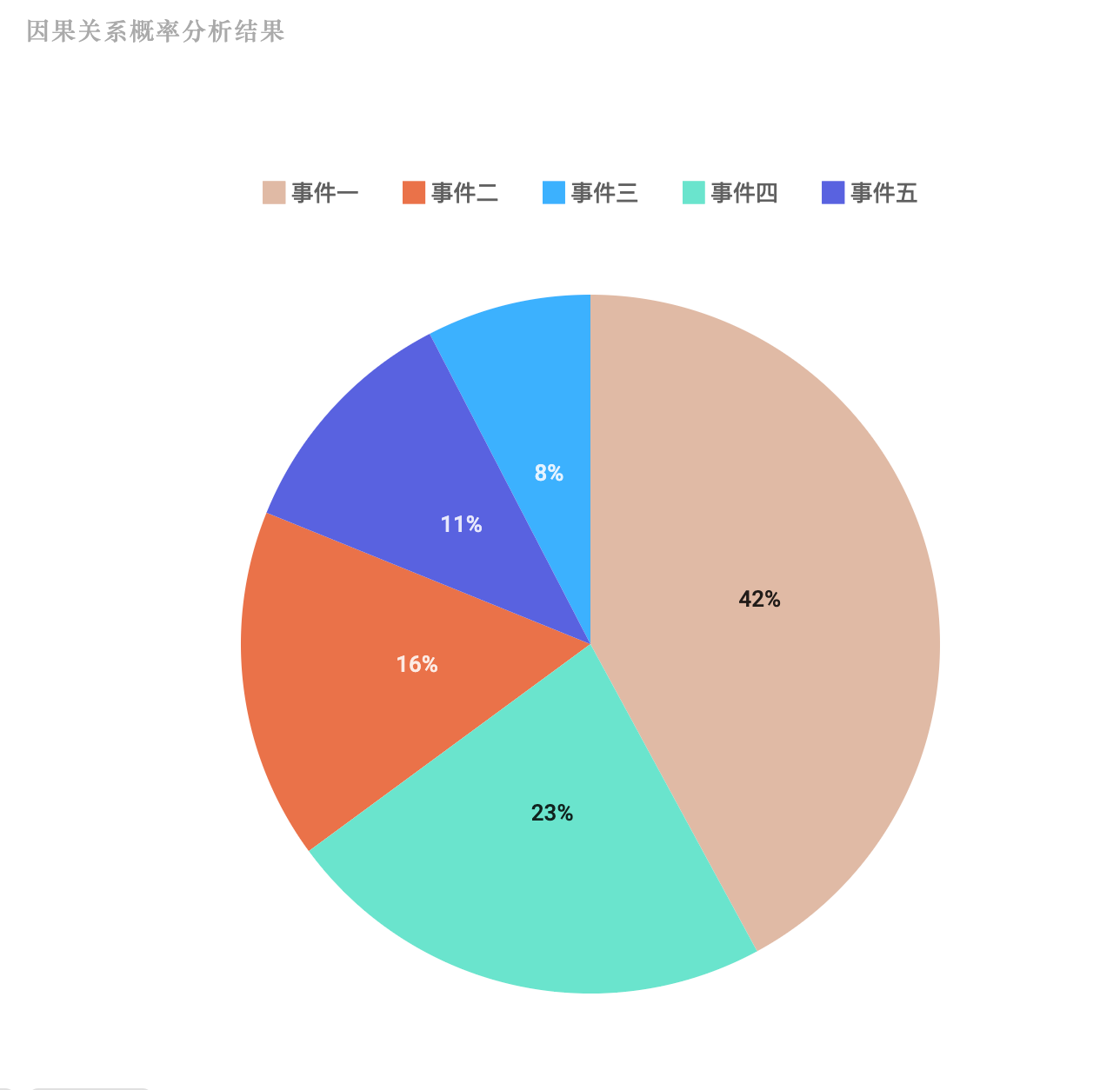


图6 模糊树分析结果页面

模糊树分析后也可再进行推理能力更强的贝叶斯网络分析，贝叶斯网络分析图如下：

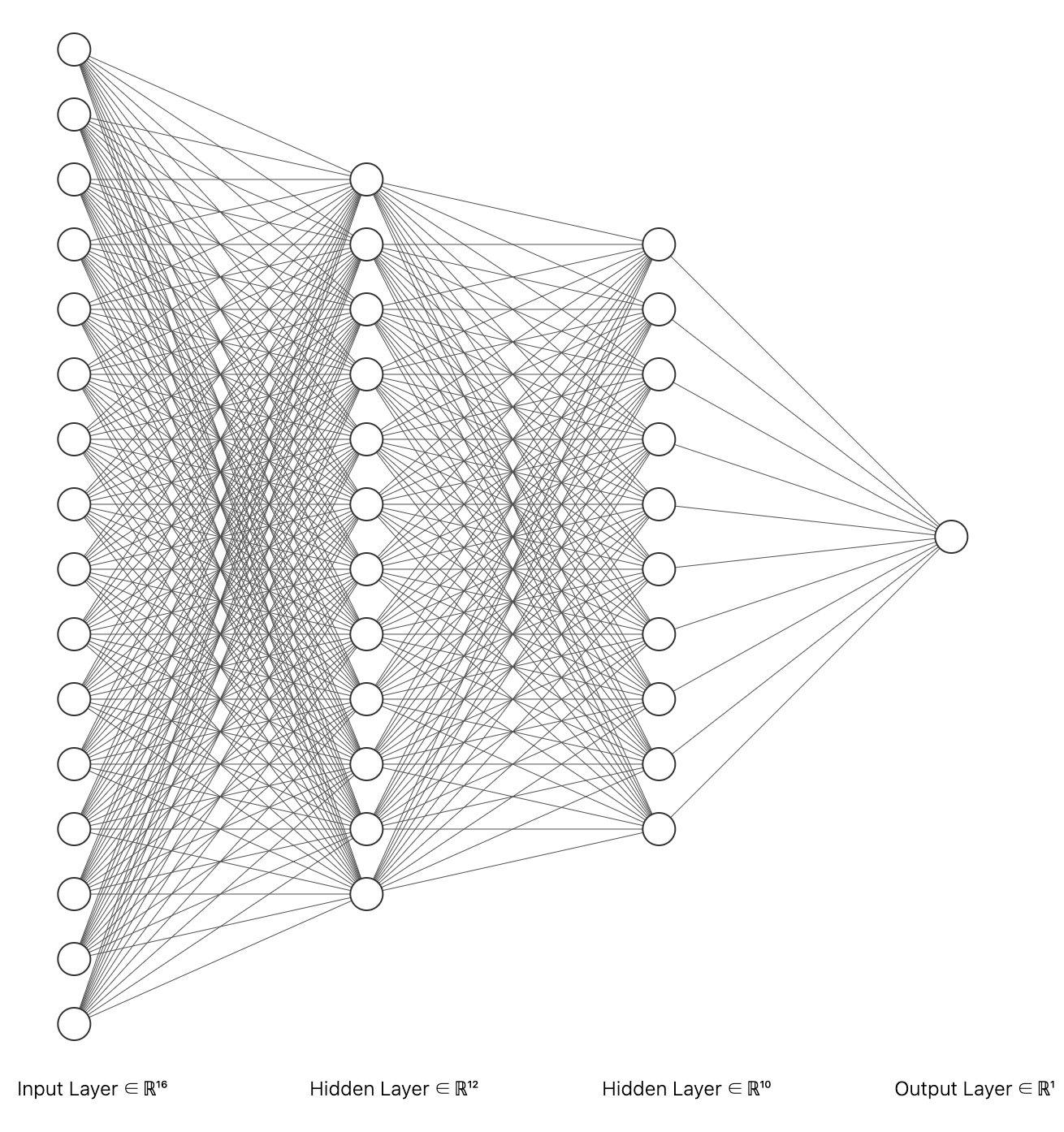


图7贝叶斯网络分析页面

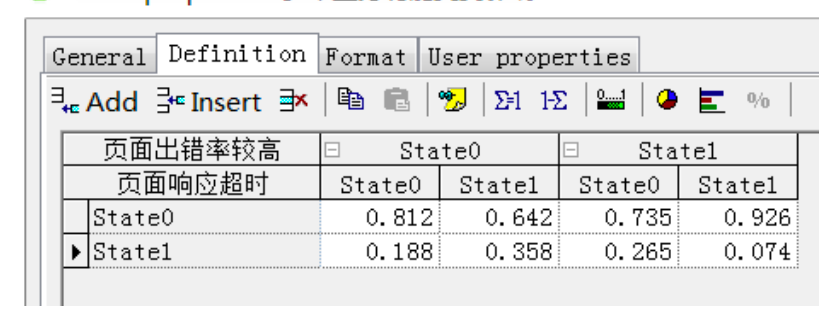


图8 贝叶斯网络分析图

此为分析State1与State2的关联分析，分析出不同State影响页面出错率与响应超时的概率。

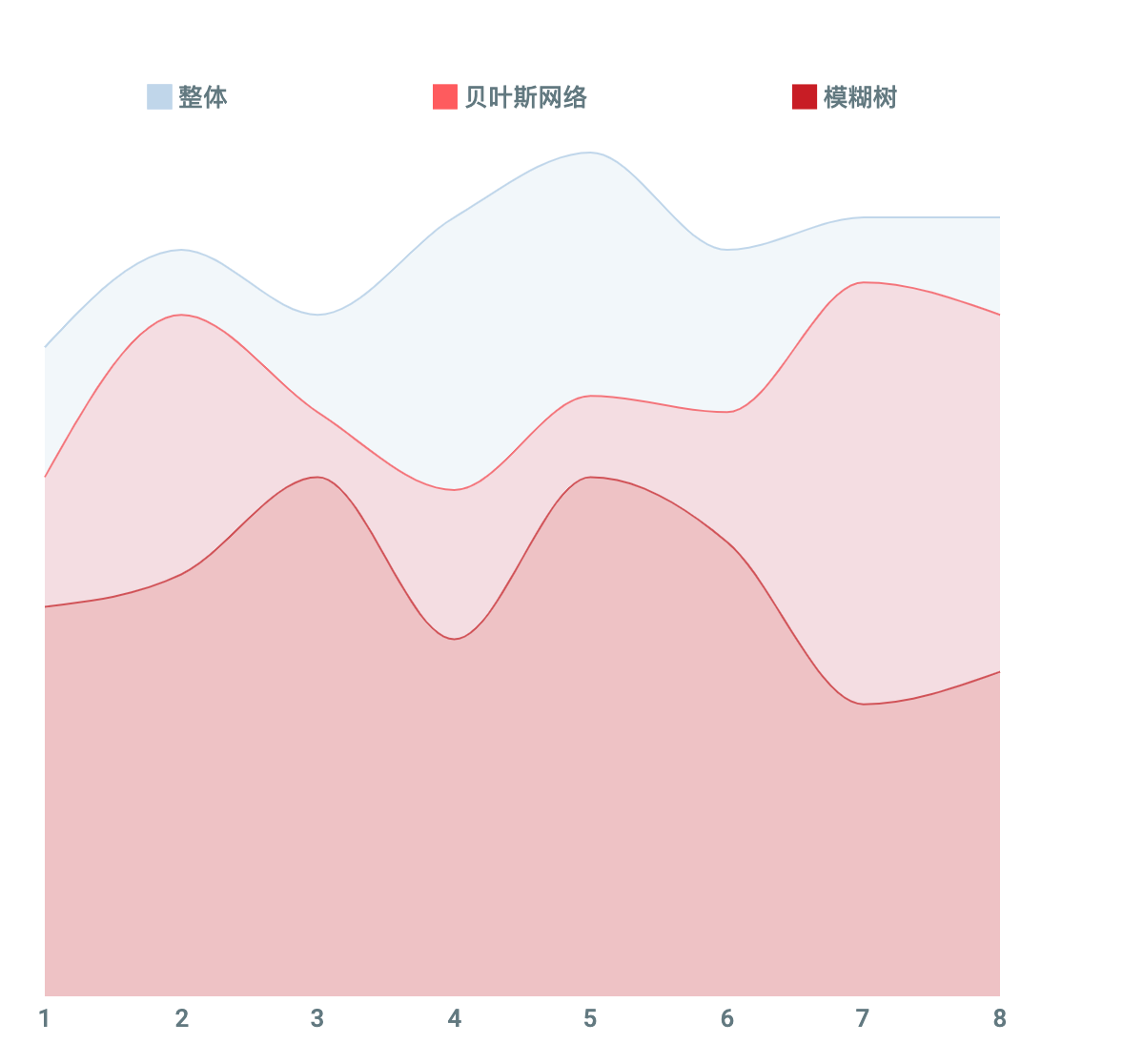


图9 整体效果对比图

事件分析后可整体查看，可以看出贝叶斯网络分析能力要优于模糊树。