四川大学

学士学位论文

基于频繁模式挖掘的网络异常事件检测系统

学生姓名：李盛乐

学科专业：计算机科学与技术

指导导师：周颖杰

完成时间：哈哈哈

SiChuan University

A dissertation for bachelor's degree

Network abnormal event detection system based on frequent pattern mining

Student’s Name:Shengle Li

Speciality:Computer science and technology

Supervisor:Yingjie Zhou

Finished time:emmmm…

摘要：随着互联网时代的到来，特别是移动互联网的高速发展，web服务和微服务已经深入到人们生活的衣食住行，人们使用互联网进行搜索、购物、支付、娱乐等等。因此，保障web服务的稳定运行变得十分重要。

Web服务的稳定运行主要靠运维人员来保障，运维人员通过监控各个关键性能指标（KPI）来判断web服务是否发生异常，KPI 异常的发生往往意味着相关的应用发生了异常。面对大量的KPI数据，亟需通过高效的网络异常事件检测系统帮助运维人员检测网络异常事件。本论文就围绕着如何通过频繁模式挖掘的算法建立一个网络异常事件检测系统，辅助运维人员对网络异常的检测。其中包括对频繁模式挖掘算法的研究和对网络异常事件检测系统的设计研究，其主要研究内容和贡献包括：

1. **对频繁模式挖掘算法的研究和使用**

本论文使用的频繁模式挖掘算法为经典的Apriori算法，Apriori算法的核心思想是通过候选集生成和情节的向下检测两个阶段来挖掘产生频繁项集，在算法执行的过程中伴随着剪枝的操作，最后通过最小支持度得到频繁项集。

1. **网络异常事件检测系统的设计和实现**

本毕业设计最终设计和实现一个网络异常事件检测系统，该系统的主要功能包括：（1）选择训练集和测试集数据；（2）绘出指定KPI的曲线图和异常点，以及异常的占比；（3）对训练集进行训练后，可选择不同类型的KPI进行检测；（4）计算并展示检测的精确率和召回率。

关键词：频繁模式；Apriori；数据挖掘；网络异常；关键性能指标（KPI）

Abstract: With the arrival of the Internet age, especially the rapid development of mobile Internet, web services and micro-services have gone deep into people's lives. People use the Internet for search, shopping, payment, entertainment and so on. Therefore, it is very important to ensure the stable operation of web services.

The stable operation of web services is mainly guaranteed by the operation and maintenance personnel. The operation and maintenance personnel determine whether the web service is abnormal by monitoring each key performance indicator (KPI). The occurrence of an abnormal KPI often means that the related application has an exception. Faced with a large number of KPI data, it is urgent to use an efficient network anomaly event detection system to help operation and maintenance personnel detect network anomalies. This dissertation focuses on how to establish a network anomaly event detection system through the algorithm of frequent pattern mining to assist network operators in detecting network anomalies. Including the study of frequent pattern mining algorithms and the design of network anomaly event detection system, its main research contents and contributions include:

1. Research and use of frequent pattern mining algorithms

The frequent pattern mining algorithm used in this dissertation is the classic Apriori algorithm. The core idea of Apriori algorithm is to mine frequent item sets through the generation of candidate set generation and the downward detection of the plot. The algorithm is accompanied by pruning during the execution of the algorithm. The operation finally obtains frequent item sets with minimum support.

1. Design and implementation of network anomaly event detection system

The graduation design finally designs and implements a network anomaly event detection system. The main functions of the system include: (1) selecting training set and test set data; (2) plotting the specified KPI graphs and abnormal points, and accounting for abnormalities (3) After training the training set, different types of KPIs can be selected for detection; (4) Calculate and display the precision and recall rate of the test.

Key words: Frequent mode; Apriori; Data mining; Network abnormal; Key performance indicators

1. 绪论
   1. 研究背景及意义

随着计算机及网络技术的发展，网络服务逐渐发展到社会的各个领域，特别是互联网及移动互联网时代的到来，web服务更是与人们的衣食住行息息相关，人们使用互联网购物、搜索、支付和娱乐等等，都离不开web服务。新中国四大发明高铁、移动支付、共享单车和网购，其中就有三个和web服务紧密相关，如今的社会没有web服务是难以想象的。因此，保障web服务稳定的运行十分的重要。与此同时，针对网络的流量攻击威胁问题也愈发严重，分布式拒绝服务攻击（Distributed DenialofService，DDoS）、僵尸网络和蠕虫攻击等频繁发生，对网络的正常运行造成极大的危害。如何在高速网络环境下对网络异常行为进行及时感知和快速处理，对于保证网络有效运行和提高服务提供能力的稳健性具有非常重要的意义[1] 。

目前的网络异常检测手段主要是依靠运维人员实时监控各种各样的关键性能指标（KPI）来判断服务是否稳定，通过判断KPI的异常来判断与其相关的应用是否发生异常。KPI 数据可以分为服务KPI和机器KPI，如今应用服务每天需要处理海量的服务请求，由此产生的KPI数据十分庞大，面对大量的网络KPI数据，仅靠人工监控以无法满足对服务异常检测的需求。而如今大数据挖掘和分析技术以十分成熟，在交通、金融、天气等方面均有成功的应用，结合大数据技术设计网络异常检测算法有很大的研究意义。

网络异常检测是指通过算法挖掘和分析KPI的时间序列数据，判断其是否出现了异常的行为。目前通过算法分析检测异常的主要难点有：

1.异常事件发生的频率很低。在实际的web应用场景中，业务系统发生异常的时刻很好，因此可用于挖掘和分析的异常数据很少；

2.异常种类繁多。在实际的业务系统中，应用十分复杂，而且处于不断更新状态，所以运行期间产生的故障类型很多，导致异常的类型也多种多样；

3.KPI的多样性。KPI有多种表现的型式，有周期型、稳定型、不稳定型和持续波动型等；

4.正常和异常的界限不是十分的精确，某些KPI服务的正常和异常的界限难以划分，导致对异常的判断十分困难。

这些难点导致现有的检测算法准确率和召回率都不高，不仅没能真正的帮助运维人员检测异常，反而因为误报和漏报增加了运维人员的工作量。所以研究和设计一个高效的网络异常检测算法十分有必要。

* 1. 研究现状
     1. 异常检测算法现状
        1. 基于分类的检测算法

基于分类的异常检测算法可以通过标签的种类分为两类：多分类问题和单分类问题。以下简单介绍各种基于分类的异常检测算法：

1. 基于神经网络

基于神经网络的算法在单分类和多分类中均可使用，多分类的基于神经网络的异常检测技术的步骤分为两步：（1）通过训练集的数据训练一个神经网络来学习不同的分类；（2）对测试集的每个数据进行测试，检验是否正常。

1. 基于贝叶斯网络

基于贝叶斯网络的技术主要应用于多分类的异常检测中。使用朴素贝叶斯网络来估计所给的数据类的标签的后验概率来判断一个单特征分类数据集，后延概率最大的标签类作为所测数据的标签类。出现零概率的数据则通常采用普拉普斯平滑解决。还可以通过汇总每个苏醒的后验概率来推广到多特征数据集。

1. 基于支持向量机

基于支持向量机是一种二分类模型，通常用于单分类的数据集中。它基础模型是定义在特征空间上的间隔最大的线性分类器，间隔最大使它有别于感应机，而支持核技巧的向量机使它成为线性分类器。在测试数据中，如果测试集数据在学习生成的区域之外则为异常点。

1. 基于规则

基于规则的技术需要学习定义异常点行为的规则。如果测试数据不包含在学习出来的任何一个规则中则被定义为异常点。通常的步骤分为两步：（1）通过训练集挖掘异常模式的频繁项集，且这些频繁项集在正常模式中不频繁出现；（2）测试集中的符合频繁项集的数据判断为异常点。

基于分类的检测算法有以下优缺点：（1）可以处理多分类问题，在测试阶段速度较快；（2）多分类问题需要精确的分类标签，直接将一个准确的标签赋给测试数据有时候不准确。

* + - 1. 基于最近邻的异常检测算法

基于最近邻的异常检测技术的原理是通过度量对象之间的距离，将距离大部分对象比较远的点判断为异常点。以下是几种常用的最近邻的异常检测算法：

1. 基于到K个近邻的距离

通过计算每个点到与其距离最近的K个点的距离，然后累加起来，异常点通常是距离大部分数据比较远的点，这就是K近邻方法。

1. 基于相对密度

以一个数据为中心，寻找能够包含K个近邻的最小超平面，求其密度，与其临近的点的密度相似则为正常点，否则为异常点。

基于最近邻的异常检测算法的优缺点：（1）适应于多种数据类型，只需要定义适合的距离即可；（2）计算复杂度较高，难以定义合适的距离。

* + - 1. 基于聚类的异常检测技术

基于聚类的算法通过设置小聚类簇的大小阈值、聚类簇距离阈值，舍弃远离其他聚类簇的小聚类簇，与其他对象非强相关的对象为异常对象。常用的聚类方法有K-means、EM、层次聚类算法等。

基于聚类的异常检测的优缺点：（1）通过数据挖掘聚类的方法，在样本充足的情况下准确度相对较高；（2）训练过程中的计算量较大，原理复杂；（3）需要通过充足的训练样本建立适当的模型；（4）如果异常数据单独成簇，则难以发现异常。

* + - 1. 统计异常检测技术

统计异常检测技术的思想是正常点出现在随机模型高概率区域，而异常模型出现在随机模型的低概率区域。常用的方法有参数化方法和非参数化方法，其中参数化方法有：（1）基于高斯模型；（2）基于回归模型；（3）基于混合参数分布。非参数化方法有：（1）基于直方图；（2）基于核函数；（3）基于密度估计函数。

统计异常检测技术的优缺点有：（1）如果数据分布的假设成立，会有统计上的异常检测方案；如果估计分布阶段对异常足够鲁棒，可以用无监督方式；（2）数据分布的假设难以确保成立，直方图难以发现属性间的关系。

* + - 1. 信息理论异常检测技术

信息理论异常检测技术的基本思想是数据中的异常会引起数据集信息内容的不规则性，通过一个数据的存在与否是否影响数据的规则性进行判断数据是否属于异常。

信息理论异常检测技术的优缺点有：（1）不需要对数据的分布做出假设；（2）效果的好坏取决于评估方法，难以得到数据的异常程度。

* + - 1. 谱异常检测技术

谱异常检测技术的基本思想是将数据嵌入到一个较低维的空间中，在此空间中正常情况和异常情况显得不同。通过寻找子空间将异常信息识别。其优缺点有：（1）时间复杂度较大；（2）可降维度，适用于高维的数据中。

* + - 1. 上下文异常检测技术

上下文异常检测技术的步骤分为两步，首先对每条数据利用上下文属性识别其上下文，然后已知的点的异常检测算法计算每条数据的异常得分。

上下文异常检测技术的优缺点有：（1）可利用实际异常的定义来检测异常；（2）只有上下文相关的属性定义后，相关技术才有效。

上述中的一些算法在网络异常事件检测中已有一定的研究和应用，但由于网络异常的种类多种多样且变化十分迅速，面对不同类型的网络异常，其中的某种算法难以达到较高的精确率和召回率，所以需要研究更加高效准确的算法用于网络异常检测系统中。

* + 1. 频繁模式挖掘算法现状

频繁模式挖掘是指通过算法挖掘出频繁出现在数据集中的模式，通过支持度和置信度找出频繁项集，从而找到潜在的关联规则，利用关联规则对测试数据进行测试。目前频繁模式挖掘算法多种多样，典型的算法在后文会详细介绍，频繁模式挖掘算法已在图像研究领域、文本分类、网络入侵检测技术、信息检索查询和web推荐系统等领域中有所应用。

在网络异常领域中频繁模式挖掘算法研究和应用还较少，没有很好的研究成果，频繁模式挖掘算法通过挖掘数据中的频繁模式，并且这些模式在正常标签中不频繁出现，通过这些频繁模式对测试集进行预测，理论上能达到较好的效果，所以研究出频繁模式挖掘算法在网络异常检测中的应用可行度很高，具有很好的应用前途。

* 1. 本文研究内容

基于上述的背景和研究意义，本文主要研究频繁模式挖掘算法的原理和实现过程，以及频繁模式挖掘算法在网络异常事件检测中的使用，对异常检测系统进行设计和实现。

具体的研究内容为基于频繁模式挖掘的网络异常事件检测系统，具体工作分为：

1. 数据的收集和预处理：对频繁模式挖掘算法研究需要大量的数据进行训练和测试，以得到异常的频繁模式。从竞赛官网下载的关于各种类型KPI的数据存在一定的缺失，需要对数据进行预处理补全缺失值。
2. Apriori算法的研究和实现：本研究使用经典的频繁模式挖掘算法Apriori算法，该算法在简单易学，实现上也不存在很大的困难，而且算法中的连接和剪枝能够保证较快的速度和较高的准确率。
3. Matlab软件的使用学习：matlab是一款软件，用于算法开发、数据可视化、数据分析以及数值计算的高级技术计算机语言和交互式环境，在算法开发和数据处理方面有很强大的功能。本研究使用matlab软件以便于更好的设计算法和设计系统。
4. 系统功能和界面的设计和实现：本设计最终需要完成一个网络异常事件检测系统，需要设计系统的功能、数据流和界面等，并实现其中的功能、处理数据的流向并将界面和后台实现连接起来，完成一个易于使用且简洁优美的系统。
   1. 论文结构安排

本文分为四个章节，每个章节注重介绍的内容均不同，本文的大纲以及内容简介如下：

第一章为绪论，主要叙述了网络异常检测的的研究背景和意义，以及研究的重要性，分析目前国内外的研究现状，简述目前主要的算法原理和应用情况，介绍本研究主要的研究和工作内容，最后描述论文的结构安排。

第二章主要介绍经典频繁模式挖掘算法，以及详细介绍本研究使用的Apriori算法的原理和步骤，描述研究的方法和过程，阐述研究和实验的结果和结论。

第三章主要阐述设计网络异常事件检测系统的工作过程，其中包括对设计系统进行的需求分析、系统的功能设计、系统数据处理流程和系统界面的设计和后台实现，展示最终完成的系统。

第四章是对本次毕业设计和研究工作的总结和展望，叙述本次毕业设计的收获、遇到的问题、解决的方法，以及对研究内容未来的展望。

1. 频繁模式挖掘算法研究
   1. 频繁模式挖掘算法介绍
      1. 概念

（引）频繁模式主要包括频繁项集、频繁子序列或者子结构，而频繁项集是频繁模式中最重要的研究内容，是生成关联规则的重要一步，也是其他频繁模式的依据。另一个方面，生成关联规则是频繁项集挖掘的主要目的，而频繁模式的概念也是最先在对购物篮关联规则分析中被提及的。由于从频繁项集生成关联规则比较直接，因此关联规则和频繁项集挖掘研究一般对应的是同一话题。

其中频繁模式挖掘的三种经典算法是Apriori算法、FP-Growth算法和Eclat算法，下面将对这三种算法进行更加详细的介绍和对比。

* + 1. Apriori算法

Apriori算法是一种最具有影响力的挖掘布尔关联规则频繁项集的算法。其通过设置最小支持度和最小置信度来筛选项集，所有大于最小支持度的项集称为频繁项集。

Apriori算法将关联规则挖掘任务分为两个步骤：

1. **频繁项集的产生**：第一步的目标是得到所有的满足最小支持度阈值的项集，这些项集称作为频繁项集。
2. **关联规则的产生**：第二步的目标是从第一步得到的频繁项集中提取所有置信度不小于设置阈值的规则，这些规则称为强规则。

Apriori算法的两个重要概念是：

1. 如若一个项集是频繁项集，则它的所有子集是频繁项集；
2. 如若一个项集不是频繁项集，则它的所有超级也不是频繁项集。

Apriori算法通过连接和剪枝连个步骤得到所有的频繁项集，然后通过计算频繁项集产生的规则的置信度得到所有的强规则。

* + 1. FP-Growth算法

FP-Growth算法是基于Apriori算法的，其通过将数据集存储在FP树上发现频繁项集。

FP-Growth算法分为两个步骤：

1. 构建一棵FP树。FP树是一种特殊的前缀树，由频繁项头表和项前缀树构成，其按支持度降序排序，支持度越高的频繁项离根节点越近，从而使得更多的频繁项可以共享前缀。在构造FP树的时候，算法需要两遍扫描数据库，第一遍扫描用来统计频率，第二遍扫描至考虑频繁项集。
2. 从FP树中挖掘频繁项集。第一步构造好FP树之后，即可从树中获取频繁项集，获取的步骤为：（1）从FP树获得条件模式基；（2）利用条件模式基，构建一个条件FP树；（3）重复（1）和（2），直到树包含一个元素项为止。
   * 1. Eclat算法

Eclat算法的思想是：两个K项的频繁项集经过或运算可以生成K+1项的频繁项集，而且将这两个K项频繁项集按照从小到大排序后，这两个K项频繁项集除了最后一项不一样外，之前的K-1项是一样的。

依据上述理论，Eclat算法可以扫描一次数据库，先获得一维的频繁项集，然后直接处理一维的频繁项集递归得到所有的频繁项集，无需在扫描数据库。实现该算法有两个阶段：（1）input阶段，主要功能是扫描一次数据库，数据存入onemap中，再导入到mymap中；（2）connect阶段，主要的功能是通过K维频繁项集生成K+1维频繁项集。

* + 1. 三种算法对比

三种算法中，Apriori算法的效率最低，因为Apriori算法需要多次扫描数据库，计算项集的支持度，扫描过程中消耗大量的时间；FP-Growth算法在长事物数据上表现很差，因为随着事物的增加，树的深度也增加，需要求解的子问题变得更多，因此效率会下降；Eclat算法的效率最高，但是由于Eclat算法使用递归的方法求解，所以数据量很大时，会给系统带来很大的负担，不适用与=于数据量大的情况。

* 1. Apriori算法

上一个小节中已经介绍了频繁模式挖掘算法的基本概念和其三种经典算法，基于本文的研究方向为使用Apriori的算法进行网络异常模式挖掘，本小节将详细介绍Apriori算法的相关概念和算法原理。

* + 1. 基本概念

Apriori算法涉及的概念包括支持度、置信度、项集和关联规则。具体知识如下：

1. 支持度：对于A->B，支持度P(A∩B)是指A和B同时发生的概率；
2. 置信度：置信度P(B|A)表示A发生时B发生的概率，计算公式为：；
3. 项集：项集是指数据集中属性以及属性的组合，如果事件A中包含k个元素，那么称事件A为k项集，如果A满足最小支持度阈值，则称为频繁k项集；
4. 关联规则：关联规则是反映一个事物与其他事物之间的相互依存性和关联性，是数据挖掘的一个重要技术，用于从大量数据中挖掘出有价值的数据项之间的相关关系。满足最小支持度和最小置信度阈值的关联关系称为强关联。
   * 1. 算法步骤

Apriori算法分为两步：连接和剪枝。下面介绍具体的算法步骤：

1. 扫描数据集，对每个候选项集计数C1，计算候选项集的支持度，删除支持度小于最支持度的项集得到频繁1-项集合L1；
2. 通过频繁K-项集生成频繁K+1-项集，方法是将所有频繁K-项集按照从小到大排序，取任意两个只有最后一个元素不同的频繁K-项集求取并集得到所有的频繁K+1-候选项集，计算所有频繁K+1-候选项集的支持度，删除不满足最小支持度的候选项集得到频繁K+1-项集；
3. 重复步骤（2）直到无法产生新的频繁K-项集；
4. 挖掘关联规则：所谓关联规则就是由某个元素集推导出其他的元素集。通过上述得到的所有频繁项集，取它们的非空子集，则这些子集与去补集形成一种关联关系，如频繁项集{A,B}有A->B和B->A两种关联关系，计算每种关联关系的置信度，其中满足最小置信度的关联关系为强关联。
   * 1. 算法优缺点

Apriori算法的优点有：原理简单，实现容易。

其缺点有：在每一步产生候选项集是循环产生的组合过多，没有排除不应该参与组合的元素；每次计算项集的支持度时，都对数据库中的全部记录进行一遍扫描比较，需要很大的计算量和I/O负载。

* 1. 研究方法
     1. 数据集下载及预处理

本研究的最终目的是设计一个网络异常事件检测系统，系统核心算法的编写测试以及系统的功能测试都需要大量的数据进行训练和测试。本研究数据均从网上下载，数据来源于多个互联网公司，经过脱敏处理后供研究者使用。

由于数据均为实时记录的线上数据，因为数据存在一些缺失值需要进行预处理方能使用，本研究采用matlab的插值函数进行数据补全，该函数通过插值点最邻近的两个点的线性函数进行预测。

* + 1. Apriori算法设计与实现

其中数据训练集每个样例包含四个属性，时间戳（timestamp）、值（value）、标签（label）和KPI ID，其中需要挖掘出KPI ID、value和label之间的关联关系，即找寻异常标签的频繁模式，且这些标签在正常模式中不频繁出现。

本研究结合Apriori算法，具体实现如下：

* 1. 实验结果及结论
  2. 本章小结

本章介绍了与本设计相关的频繁模式挖掘算法，包括算法的概念、经典算法的介绍和比较，以及介绍了设计中的研究方法，包括数据的下载和预处理、Apriori算法的设计和实现，最后通过实验的结果，给出实验的结论。

1. 网络异常事件检测系统的设计和实现
   1. 需求分析

本研究为设计一个基于频繁模式挖掘的网络异常事件检测系统，通过分析和调研，发现系统最好可以数据可视化、结果可视化和选择不同类型的KPI进行检测等功能，按照系统的定位和调研结果，得出了以下的需求分析。

首先用户可以通过系统选择需要检测的KPI数据对应的训练集和测试集。为满足用户的需求，本系统对KPI的检测不限定于某些类型，而是可以选择多种不同类型的KPI进行检测。

选择了数据文件后，系统会读取文件的内容，并识别出其中包含的KPI种类，以列表的形式将所有的KPI ID展示给用户，用户可以通过绘图功能选择指定的KPI，将其以时间戳为横坐标，value为纵坐标绘出曲线图，并标出其中的异常点，并计算该KPI中异常所占的比例，以饼图形式展示。此功能可以让用户更加直观的查看KPI的模式，以及查看异常发生的情况和比例。

最后是最重要的功能，即对KPI进行检测。用户可以通过检测功能对读取的测试数据的KPI进行检测，用户可以选择不同类型的KPI进行检测。系统检测完成后，同样会将测试数据的KPI曲线图绘画出来，同时展示原始异常和算法预测异常的对比，计算系统预测的精确率和召回率，以可视化的形式展示在用户面前。

* 1. 系统功能设计

系统功能图如下所示：

用户界面

功能接口

读文件功能

绘图功能

检测功能

导出文件功能

文件读取服务

数据解析服务

数据存储服务

绘图服务

异常计算服务

异常模式挖掘服务

测试集检测服务

文件导出服务

图1 系统功能设计图

* + 1. 读取文件

用户点击文件按钮即可选择测试集和训练集，待系统读取文件完毕后，系统将用户选择的文件显示在界面。

* + 1. 绘画KPI曲线图

系统读取用户数据后，会从训练集中找出KPI ID的种类，并将ID的列表展示在界面，用户可以选择想要查看的KPI，点击绘图功能，即可将该KPI的曲线图绘制在界面上，并且会标出其中的异常点，同时以饼图的方式展示异常所占的比例。

* + 1. 异常频繁模式挖掘

选择文件之后，用户即可使用系统对训练集训练，只需选择训练功能，系统会自动开始挖掘异常标签中的频繁模式。

* + 1. 测试及评估算法

进过训练数据后，得到异常标签的频繁模式，用户即可使用挖掘的模式对测试集进行测试，系统测试完毕后，同样会将测试的异常点以曲线图的形式展示给用户，同时展示原始的数据异常图便于用户对比。并计算检测结果的精确率和召回率。

* + 1. 导出测试集

对测试集测试完毕后，用户可将测试的结果导出到文件中，以便后续查看结果时直接查看。

* 1. 数据流图

用户导入的数据流转图如下所示：

用户

训练集和测试集

KPI ID

KPI数据文件

KPI ID

KPI数据

数据图形

KPI ID

KPI数据

检测结果

图2 系统数据流图

* 1. 系统功能实现
     1. 系统界面设计

按照系统的需求分析和功能设计，系统界面以简洁实用为原则，在保证所有需求实现的同时，给用户带来良好的实用体验。系统界面使用matlab GUI完成。

图图图图图

图3给出了系统界面设计图。该界面分为三个模块，分别为顶端的菜单栏、中部的文件选择和绘图模块以及底部的KPI检测模块。

菜单栏为用户提供的功能包括File、Edit、Tool和Help菜单。功能如下：

1. File菜单提供了打开文件、保存文件和退出程序的功能；
2. Edit菜单提供了清除数据、选择曲线图颜色和缩放图片的功能；
3. Tool菜单栏提供了图片下载、测试结果导出功能；
4. Help菜单栏提供了软件介绍和使用帮助的功能。

中部文件选择和绘图模块主要的功能为：

1. 可用通过菜单栏或者文件选择按钮选择需要读取的训练集和测试集；
2. 文件选择完毕后，系统会展示训练集和测试集的文件名以便用户查看；用户选择文件后，系统自动读取文件的数据，并将数据保存到工作空间以便后续操作；
   * 1. 系统后台实现
   1. 设计结果
   2. 本章小结
3. 总结与展望
   1. 总结
   2. 展望

参考文献：

1. 郭通. 基于自适应流抽样测量的网络异常检测技术研究[D]. 中国人民解放军信息工程大学, 解放军信息工程大学, 2013.