1.重复检测函数

重复检测函数的主要特点是:

高质量的——质量是匹配的特点,我们重复检测特性提供了这个承诺。

可伸缩的——大型数据集重复检测利用Hadoop框架几乎无限的可伸缩性。

快速和互动在单个机器上——你可以快速、交互地完善您的重复检测模型。

国际——国际数据支持和没有区域知识编码成重复数据删除引擎——你提供外部的业务规则。

基于机器学习的——重复检测引擎配置示例。在一系列的训练可以改进重复数据删除模型简单地通过对话与工具是什么,什么不是复制的一个很好的例子。

提示：

重复检测是否适合使用原始数据。但是如果你有脏数据和数据注册方式有很大的差异,我们建议你首先尽力标准化之前找到重复的数据。标准化可以通过修剪,分词,去掉不需要的字符,替换同义词之类的东西

2.字符集分布（character set distribution analyze）分析根据字符集关联来检查和映射文本字符，比如拉丁文，希伯来语，西里尔语，汉语等等；

3.参考数据匹配器（reference data matcher）：The 'Reference data matcher' analyzer provides an easy means to match several columns against several dictionaries and/or several string patterns. The result is a matrix of match information for all columns and all matched resources.

对一些字典 和/或 几个字符串模式匹配多个列。结果是一个所有列和匹配资源的匹配信息矩阵。

4.参照完整性（Referential integrity）：检查记录之间的关键关系是否完整，单表，多表，多个数据存储的表等的关系。（功能：字典表参照校验，以一张表为字典表，两张表之间进行匹配，未匹配上的高亮显示与图中标红处，右上角：save dataset

通过一个配置文件的元素, conf.xml,这样就可以改变DataCleaner的静态配置和配置环境。配置文件和相关的文件都存储在一个文件夹按照惯例 .datacleaner在用户的主目录。

DataCleaner监视web应用程序中,conf.xml文件是唯一的配置。该文件位于每个租户的存储库的根文件夹。

在配置文件中大部分的元素也可编辑在桌面应用程序中。然而重要的是要注意,更改在GUI并不直接保存到配置文件,但到userpreferences.dat文件。您可以考虑两个文件之间的关系:配置文件定义了一个静态的,unmodifyable原型的应用程序环境。所有定制这个原型userpreferences保存在桌面应用程序。dat文件。

5. 数据存储在配置文件中可以配置在< datastore-catalog >元素。以下部分将进入进一步的细节与特定类型的数据存储。

6. 数据库(JDBC)连接

这里有一些常见的数据库类型的例子。

MySQL

<jdbc-datastore name="MySQL datastore"> <url>jdbc:mysql://hostname:3306/database?defaultFetchSize=-2147483648</url> <driver>com.mysql.jdbc.Driver</driver> <username>username</username> <password>password</password> <multiple-connections>true</multiple-connections> </jdbc-datastore>

1. 完整性 Completeness：完整性用于度量哪些数据丢失了或者哪些数据不可用。

2. 规范性 Conformity：规范性用于度量哪些数据未按统一格式存储。

3. 一致性 Consistency：一致性用于度量哪些数据的值在信息含义上是冲突的。

4. 准确性 Accuracy：准确性用于度量哪些数据和信息是不正确的，或者数据是超期的。

5. 唯一性 Uniqueness：唯一性用于度量哪些数据是重复数据或者数据的哪些属性是重复的。

6. 关联性 Integration：关联性用于度量哪些关联的数据缺失或者未建立索引。

注意DataCleaner只是一个数据质量分析工具，而不是一个数据分析工具，只专注于数据质量问题的分析，而不是数据分析。这些DataCleaner提供的分析主要还是帮助我们来衡量数据的的完整性Completeness、规范性 Conformity、 一致性 Consistency、准确性 Accuracy、 唯一性 Uniqueness、关联性 Integration