|  |
| --- |
| **常用数据匹配算法简介系类** |
| --Jaro–Winkler distance 算法 |
|  |
| 数据是组织最具价值的资产之一。企业的数据质量与业务绩效之间存在着直接联系，高质量的数据可以使公司保持竞争力并在经济动荡时期立于不败之地。本文介绍在数据匹配算法 Jaro–Winkler distance |
|  |
| **wanghetommy@163.com** |
| **2011-6-8** |
|  |

目 录

[About this document 4](#_Toc295291475)

[Jaro- Winkler Distance 4](#_Toc295291476)

[1) Introduction 4](#_Toc295291477)

[2) Definition 4](#_Toc295291478)

[a) The Jaro distance算法最后得分公式: 4](#_Toc295291479)

[b) Match Window(匹配窗口)计算公式 4](#_Toc295291480)

[c) 解释上面两组公式 5](#_Toc295291481)

[d) Jaro–Winkler distance算法公式 5](#_Toc295291482)

[3) 引用资源 5](#_Toc295291483)

|  |  |  |
| --- | --- | --- |
| 教程版本 | 说明 | 作者 |
| 0.9 | 总结常用ETL数据匹配算法理论 | wanghe |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# About this document

数据是组织最具价值的资产之一。企业的数据质量与业务绩效之间存在着直接联系，高质量的数据可以使公司保持竞争力并在经济动荡时期立于不败之地。有了普遍深入的数据质量，企业在任何时候都可以信任满足所有需求的所有数据。然而大多数情况下、我们的数据源来自企业中各个系统、并不是所有的数据都有统一的标准、这样在我们做ETL过程的时候就需要将我们需要的数据清洗出来、而清洗的过程中主要是对数据进行操作、因此采用一个高效的合理的算法必不可少。此文档简单对数据匹配的几种经典进行阐述与分析。以供参考，此文档介绍Jaro–Winkler distance。

# Jaro- Winkler Distance

1. Introduction

The Jaro–Winkler distance (Winkler, 1990)是计算2个字符串之间相似度的一种算法。它是Jaro distance算法的变种。主要用于record linkage/数据连接（duplicate detection/重复记录）方面的领域，Jaro–Winkler distance最后得分越高说明相似度越大。Jaro–Winkler distance 是适合于串比如名字这样较短的字符之间计算相似度。0分表示没有任何相似度，1分则代表完全匹配。

1. Definition

### The Jaro distance算法最后得分公式:

***; //公式一***

**其中:**

* **s1、s2 是要比对的两个字符**
* **是最后得分**
* **m是匹配的字符数**
* **t 是换位的数目**

### Match Window(匹配窗口)计算公式

**; *//公式二***

**其中:**

* **s1、s2 是要比对的两个字符**
* **MW是匹配窗口值**

### 解释上面两组公式

1.字符串s1与字符串s2在做匹配计算时，当两个字符的距离不大于公式二的最后结果(匹配窗口)即认为是匹配的。

2.当s1、s2中字符相匹配但是字符位置不一样时发生换位操作、而公式一中换位的数目t为不同顺序的匹配字符的数目的一半。比如:两个字符串*CRATE*和*TRACE*做匹配操作，字符串中仅有'R' 'A' 'E'三个字符是匹配的，即m=3。为什么'C', 'T'不算做是匹配的呢。因为虽然'C', 'T'都出现在两个字符串中，但是通过公式二得出匹配窗口值为 (5/2)-1=1.5。而两个字符串中'C', 'T'字符的距离均大于1.5。所以不算做匹配。因此t=0。在另一组字符串*DwAyNE* 与 *DuANE* 。匹配的字符D-A-N-E 在两个字符串中有相同的字符顺序，所以不需要进行换位操作，因此t=0,m=4。

### Jaro–Winkler distance算法公式

Jaro-Winkler算法给予了起始部分就相同的字符串更高的分数，它定义了一个前缀范围p，对于要匹配的两个字符串，如果前缀部分有长度为L的部分字符串相同，则Jaro-Winkler Distance为:

; ***//公式三***

**其中:**

* **是Jaro distance最后得分**
* **L是前缀部分匹配的长度**
* **P是一个范围因子常量，用来调整前缀匹配的权值，但是P的值不能超过0.25，因为这样最后得分可能超过1分.Winkler的标准默认设置值P=0.1。**

Although often referred to as a *distance metric*, the Jaro–Winkler distance is actually not a [metric](http://en.wikipedia.org/wiki/Metric_%28mathematics%29) in the mathematical sense of that term.

1. Example

### 例子1

给出两个字符串 *s1 MARTHA* 和 *s2 MARHTA*、我们可以得出：

* *m* = 6
* | *s*1 | = 6
* | *s*2 | = 6
* 两组字符T/H和H/T要进行换位操作，因此t=2/2=1;

我们可以根据公式一得出Jaro得分：

如果使用Jaro–Winkler，并且取范围因子P=0.1,我们会得出:

P=0.1

L=3

假使串 *s*1 *DWAYNE* 并且 *s*2 *DUANE* 我们发现：

* *m* = 4
* | *s*1 | = 6
* | *s*2 | = 5
* *t* = 0

我们发现一个Jaro比分：

使用标准砝码发现Jaro-Winkler比分 *d* = 0.1我们继续发现：

因而：

*dw* = 0.822 + (1 \* 0.1 (1 − 0.822)) = 0.84

### 例子2

给出两个字符串 *s1 DWAYNE and s2 DUANE*、我们可以得出：

* *m = 4*
* *| s1 | = 6*
* *| s2 | = 5*
* *t = 0*

我们可以根据公式一得出Jaro得分：

如果使用Jaro–Winkler，并且取范围因子P=0.1,我们会得出:

P=0.1

L=1

### 例子3

在这个例子中我们利用图形来说明匹配过程，给出两个字符串

*s1 DIXON and s2 DICKSONX* 得出下面图表：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | D | I | X | O | N |
| D | 1 | 0 | 0 | 0 |  |
| I | 0 | 1 | 0 | 0 | 0 |
| C | 0 | 0 | 0 | 0 | 0 |
| K | 0 | 0 | 0 | 0 | 0 |
| S |  | 0 | 0 | 0 | 0 |
| O |  |  | 0 | 1 | 0 |
| N |  |  |  | 0 | 1 |
| X |  |  |  |  | 0 |

其中：1代表匹配命中，0代表无匹配，匹配窗口mw = 8/2 -1 =3.▄颜色块代表匹配范围，即在匹配窗口之内。

根据图标我们得出：

* *m = 4 虽然两个字符串都含有字符‘X’但是距离已经超过匹配窗口值，所以忽略*
* *| s1 | = 5*
* *| s2 | = 8*
* *t = 0*

我们可以根据公式一得出Jaro得分：

如果使用Jaro–Winkler，并且取范围因子P=0.1,我们会得出:

P=0.1

L=2

1. 引用资源

[Wiki百科](http://en.wikipedia.org/wiki/Jaro-Winkler_distance)