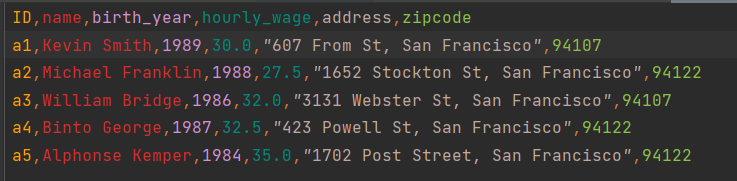
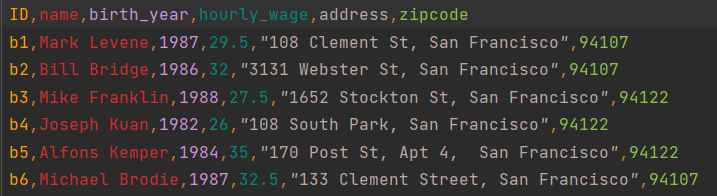
数据集要求具有相同模式，如下所示：

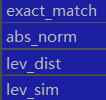
输入的数据集A



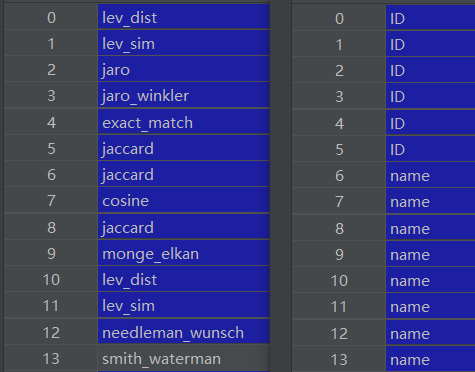
输入的数据集B：

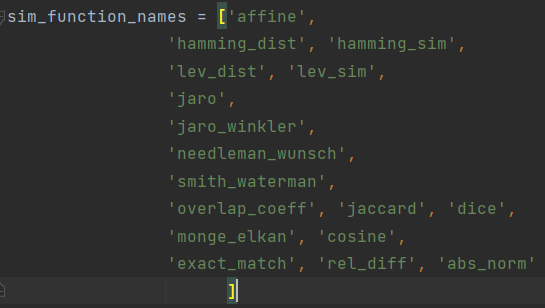


其中，birth\_year,hourly\_wage,zipcode中的数据是数字类型的，解析时处理方式相同，采取了如下四种相似度函数：



对于字符型数据，采取了以下相似度函数



该框架实际上提供了16种相似度函数在py\_entitymatching\feature\simfunctions.py，如下所示：  


Affine（neddleman\_wunsch一种）:处理data gap的数据。即存在缺失值，数据误解（Data misinterpretation），数据差异（Data disparity）。使得数据更加平滑。

汉明距离/相似度：

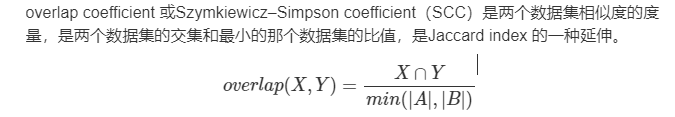
莱温斯坦距离/相似度：编辑距离的一种

Jaro ，Jaro–Winkler距离/相似度：[(81条消息) 字符串相似度比较算法：Jaro–Winkler similarity的原理及实现\_程序员的自我反思的博客-CSDN博客\_jaro相似度](https://blog.csdn.net/a553181867/article/details/89057661)

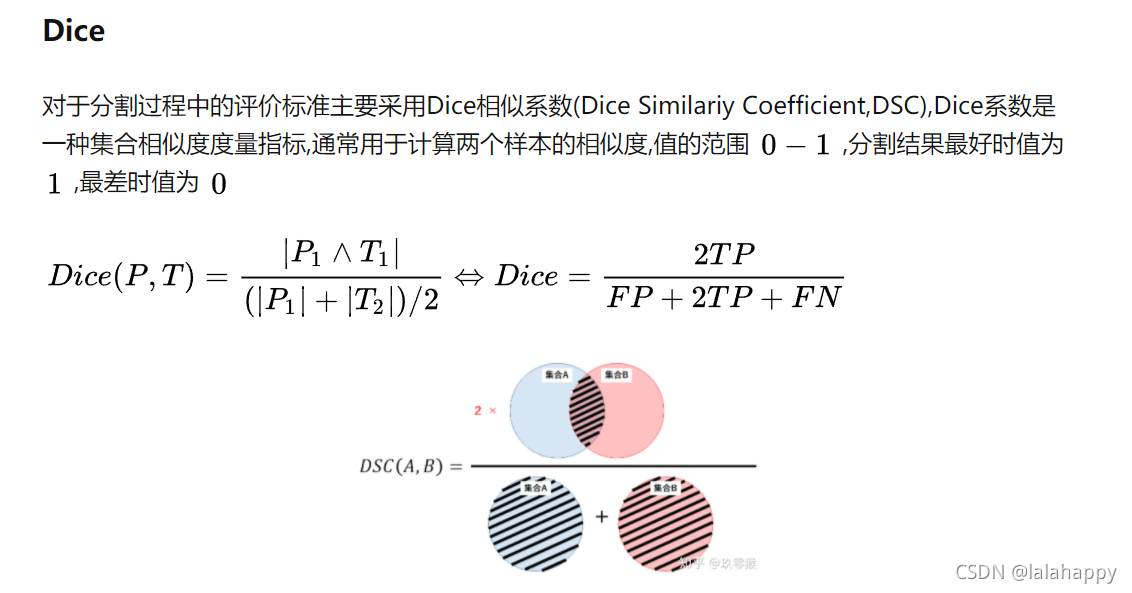
neddleman\_wunsch：(dp算法)是基于生物信息学的知识来匹配蛋白序列或者DNA序列的算法。效地解决了如何将一个庞大的数学问题分解为一系列小问题，并且从一系列小问题的解决方法重建大问题的解决方法的过程。广泛应用于优化整体序列比较中。

Smith\_waterman：(dp算法)找出两个序列中具有高相似度的片段

Overlap\_coeff：



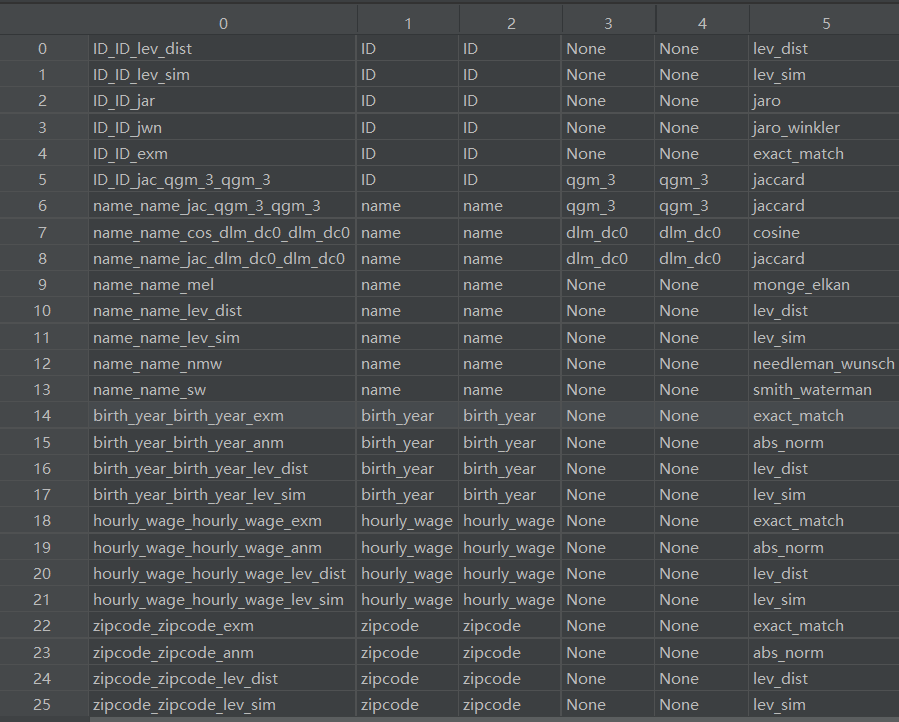
Dice：



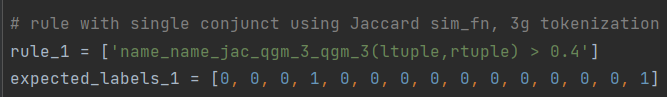
Monge\_elkan: 结合了基于序列和基于集合的方法的优点。这对于需要对相似性度量进行更多控制的领域可能更有效。

Rel\_diff：relative difference。衡量一个数据集中的语言特征（主题，短语，单词，情感或语法元素）与其他数据集相比的流行程度的指标。

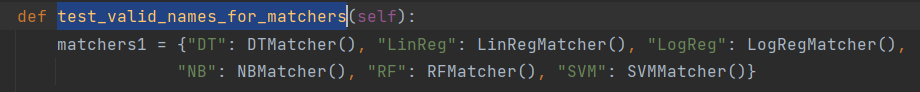
分词方法：q-gram（q=3）或者分隔符delimiter。此外，该框架提供了空格分词，最大长度字母分词（提取最长连续字母序列），最大长度字母数字分词（提取最长连续字母数字序列）的分词方法。

特征空间构建的Feature table：  


Matcher：  
1. 基于规则：



2. 机器学习：



代码由sklearn实现