Chapter 3

Entity Resolution on Names

ABSTRACT

Errors with names occur frequently. “California” and “CA” refer to the same state of the USA; however, they may both appear as records in a database at the same time. Several techniques need to be proposed to solve these problems. In this chapter, the authors introduce the methods of entity resolution on names. They propose three methods. Similarity measure between names is a kind of fundamental techniques; it makes a significant contribution to the textual similarity. The method of string transformations can handle some situations beyond textual similarity. Recently, learning algorithms on string transformations have been proposed to make matching robust to such variations. Examples illustrate the benefits of each approach.

INTRODUCTION

In the real world, we may be confronted with many errors about names. Record matching is a well-known problem of matching records that can handle this situation. Most approaches to record matching just rely on textual similarity of each pair record. The applications include entity resolution in E-Commerce (Chapter 16), bibliography (Chapter 15) and medical health information system (Chapter 17).

In section 1, we will introduce several metrics computed using a similarity function. Levenshtein (1966) has proposed a metric to calculate the distance between two sequences called Edit-Distance. The edit distance metrics work well for catching typographical errors, but they are typically ineffective for other types of mismatches. In Elmagarmid, Ipeirotis, and Verykios (2007), Smith and Waterman describe an extension of edit distance and affine gap distance in which mismatches at the beginning and the end of strings have lower costs than mismatches in the middle. This approach is a well-known algorithm for performing local sequence alignment, that is, for determining similar regions between two nucleotide or protein sequences. Instead of looking at the total sequence, the Smith-Waterman algorithm compares segments of all possible lengths and optimizes the similarity measure in Smith & Waterman (1981). Jaro-Winkler Distance in Jaro (1989) and Winkler (1990) introduces a string comparison algorithm that is used in the area of record linkage (duplicate detection). The higher the two strings’ Jaro-Winkler distance, the more similar the strings are. The Jaro-Winkler distance metric is designed and best suited for short strings such as person names. The above three algorithms
are character-based similarity metrics that are designed to handle typographical errors. Token-based similarity metrics are widely used in the domain of information retrieval. Scientists have proposed TF-IDF that is a numerical statistic which reflects how important a word is to a document in a data collection in Manning, Raghavan, and Schütze (2008). Cosine similarity in Tan (2007) is often coming along with TF-IDF. Cohen described a system named WHIRL in Cohen (1998, June) that adapts from information retrieval the cosine similarity combined with the TF-IDF weighting scheme to compute the similarity of two fields in Elmagarmid, Ipeirotis, & Verykios (2007). Character-based and token-based similarity metrics focus on the string-based representation of the database records. However, strings may be phonetically similar even if they are not similar in a character or token level. Soundex is a phonetic algorithm for indexing name by sound, as pronounced in English. The goal is for homophones to be encoded to the same representation so that they can be matched despite minor differences in spellings in R.C. Russell Index and R.C. Russell Index.

Record matching infrastructure does not allow a flexible way to account for synonyms that refer to the same name with different manifestations, and forms of string transformation such as abbreviations. In section 2, we will introduce a transformation-based framework for record matching. At first, we introduce preliminary knowledge about Context-Free grammar which is also referred as CFG in Hopcroft (2008). The context-free grammar is widely used in compiler theory. The parse tree that based on CFG, is one of the most efficient method to parse grammar. It can be constructed easily and it is effective to process semantic actions. Two frameworks have been proposed as transformation-based entity representation that has been shown in Arasu, Chaudhuri & Kaushik (2008) and Arasu & Kaushik (2009). In this section, we mainly introduce the grammar-based entity representation framework.

At first, the framework generates productions based the real world to construct a context free grammar. Then it utilizes the parse tree technique to analysis how a record is generated by the extension grammar of the CFG, and adds semantic actions to the parse tree in order to determine whether two records are the same. For example, “Dr Andrew J. Smith” will be analyzed as first name is Andrew, and last name is Smith. Then, “Smith, Andy J.” will also be analyzed as first name is Andrew, and last name is Smith. Therefore, this framework can solve this problem.

Recently, both machine learning and data mining algorithms are widely used in computer science. Learning algorithm is important and effective to solve the problem of entity resolution on names. S. Tejada has proposed a system on learning algorithm to handle this situation in his Ph.D thesis (in Tejada, Knoblock & Minton 2001). There are two stages in the system architecture. The stage of computing similarity scores takes advantage of the methods about textual similarity that we introduce in section 1. The next stage is mapping learner. The first part of the mapping learner is mapping rule learner that uses decision to classify whether the rules are mapped or not. The transformation weight learner computes the weight of each transformation in order to get higher accuracy by recalculating similarity scores. The experimental results show high accuracy outcome of the processing.

At last, we will list some books and papers for further work as the expansion of knowledge.

**BACKGROUND**

Reduce error or duplicate records is one of the difficult problems in data cleaning. As the number of entities increasing fast, errors or duplicate records occur in databases more frequently. Traditional methods mainly rely on textual similarity. Recently, scientists have proposed some approaches that deal with these situation with higher accuracy.