Asian Journal of Information Technology 12 (7): 236-241, 2013
ISSN: 1682-3915

Synonym Based Duplicate Record Detection

[1]K. Amshakala and [2]R. Nedunchezhian
[1]Coimbatore Institute of Technology, Coimbatore, India
[2]Sri Ramakrishna Engineering College, Coimbatore, India

Abstract: As the amount of data and data providers are increasing tremendously, there is a high demand for integrating data from heterogeneous data sources. Often, in the real world, entities have two or more representations and data are not defined in a consistent way across different data sources. When answering user's query, results are returned to the users by combining data from several databases and the results include duplicate entries. Duplicate detection techniques detect multiple representations of identical real world entities. Without using duplicate record detection techniques, the quality of the extracted data remains low. This study presents an unsupervised duplicate record detection technique which does not require expert's knowledge or hand coded rules to detect duplicate records. A large lexical database called WordNet ontology is used to match the entities.

Key words: Data integration, duplicate record detection, WordNet ontology, synonyms, catalog integration, un-supervised matching

INTRODUCTION

Integrating data from multiple databases results in duplicate records, since same entities have two or more representations in different databases. When the user submits a query, the server will retrieve the corresponding results from multiple data sources and presents it to the user. The query results from different data sources may refer to the same real world entity. The problem of identifying two or more records describing the same entity is of major concern in delivering correct query results to the user.

A fundamental data integration task faced by the commercial portals is the integration of product catalog of multiple online shopping dealers into a single product catalog (Papadimitriou *et al*., 2012). The single master product catalog has the same product named in different ways. For example, when the product mobile is searched, the query results returned by two online shopping portals Junglee.com of Amazon and Pricegrabber.com are different. It is found that Junglee.com refers cellular phone as mobile phone and Pricegrabber.com refers it as cell phones. Duplicate records do not share a common key or/and they contain errors that make duplicate detection a difficult task. Also, large amount of structured information is now derived from unstructured text and from web. This information is typically imprecise and noisy. The increasing popularity of information extraction techniques is going to make this issue more prevalent in the future, highlighting the need to develop robust and scalable solutions. This only adds to the sentiment that more research is needed in the area of duplicate record detection, data cleaning and information quality in general.

One of the most common sources of mismatches in database entries is the typographical variations of string data. Therefore, duplicate detection typically relies on string comparison techniques to deal with typographical variations. Multiple methods have been developed for this task and each method works well for particular types of errors. While errors might appear in numeric fields as well, the related research is still in its infancy. In most real life situations, however, the records consist of multiple fields making the duplicate detection problem much more complicated.

Exact string matching techniques only detect typographical variants of string fields and do not take into consideration, the synonyms of the string fields. For example, in the product catalog shown in Table 1 although the string matching methods could detect the similarity between data processor and processor, computer and personal computer, it could not detect the similarity between the products data processor, personal computer and desktop. In such cases, semantics based comparison helps in detecting duplicates. To compare the semantics of the string fields, ontology specific to the domain under consideration can be used. WordNet ontology is a lexical database and it is used to retrieve synonyms of the string fields to be compared.

When considering the entire record, different fields must be given different importance. Weights are assigned to different fields inorder to assess the similarity metric

Corresponding Author: K. Amshakala, Coimbatore Institute of Technology, Coimbatore, India

Table 1: Sample product catalog

ID	M_Name	P_Name	Price
101	DELL	Printer	31200
102	HP	Desktop	42300
103	HCL	Data processor	40000
104	COMPAQ	Monitor	40000
105	HCL	Hard disk	14600
106	WIPRO	Computer	36000
107	ZENITH	Fixed disk	15600
108	HP	Personal computer	40700
109	DELL	CPU	44000
111	LENOVO	Processor	45000
112	SONY	RAM	20000
113	ZENITH	Central processing unit	10000
114	TOSHIBA	Random memory	21000
119	ACER	Random access memory	21500

between the records. Attribute entropy is a good measure of structuredness of data. Regularities and structuredness are characterized by small entropy values whereas randomness is characterized by large entropy values (Yao, 2003). A duplicate record matching technique called SDRD (Synonym based Duplicate Record Detection) is proposed in this study. SDRD compares records by string matching or synonym matching for text fields and numerical approximations for numerical fields. Different weights are assigned to various fields based on the importance of the field in contributing to the matching accuracy.

PROPOSED WORK

In this study, the duplicate record detection problem is defined and this is followed by the assumptions and limitations of the proposed method. It also describes the synonym based similarity metric evaluation and the method used to group duplicate and non-duplicate records.

Problem definition: Record matching refers to the task of finding records that refer to the same entity across different data sources. Let R and S be the two data sets that are to be integrated. For each record pair (t_r, t_s) where $t_r \in R$ and $t_s \in S$, record matching classifies the pair as either matching or non-matching based on the similarity between the records. The record pair (t_r, t_s) is represented as random vector $V = [v_1, v_2, v_3 \ldots v_n]$ with components that correspond to n comparable fields of R and S. Each v_i shows the level of agreement of the ith field for the records t_r and t_s. $v_i = 1$, if the ith field of t_r and t_s agrees to match or $v_i = 0$, if the records disagree to match. A similarity metric is evaluated as the accumulated similarity between all the fields of the two records. A threshold can be fixed to classify the records as matching if the similarity metric is above the fixed threshold.

Preliminaries: The basic definitions of terms used in this study are given.

Definition 1 (Entropy and information): The information content I of a single event or message is defined as the base-2 logarithm of its probability P(x):

$$I = \log_2 (P(x)) \quad (1)$$

Entropy can be regarded as uncertainty or disorder. To gain information is to lose uncertainty by the same amount, so I and H differ only in sign (if at all): H = -I (Cover and Thomas, 1999). Entropy of a field X, H(X) is determined using:

$$H(X) = -\sum p(x) \log_2 (p(x)) \quad (2)$$

where, p(x) is the probability distribution of the values of the field X (Cover and Thomas, 1991). The attribute entropy H(X) serves as a measure of diversity or unstructuredness. It is determined by the probability distribution of the attribute in the entire population and does not depend on any other attributes (Yao *et al*., 1999).

Definition 2 (Duplicate ratio: Suppose there are m records extracted from a data source d then totally $n = m \times (m-1)$ record pairs are generated by putting every two records together. Suppose t of the n record pairs are duplicate records then the duplicate ratio of the m records is t/n (Su *et al*., 2010).

Assumptions and limitations: In this study, researchers present the assumptions and observations on which SDRD is based:

- A global integrated schema exists and each local database's schema has been matched to the global schema
- The records from the same data source usually have the same format
- Most duplicates from the same data source can be identified and removed using an exact matching method
- Products are described using their complete name and not abbreviations

Duplicate detection method: In this study, a novel record matching technique, suitable for catalog integration is proposed. The string fields of the records to be matched are compared using exact string matching and if this fails, synonyms of the strings are compared. In exact string matching approach, the string to be matched is checked if it contains the same characters in the same sequence. If the strings are exactly the same, they are identified as duplicates. In some cases, the sequence of characters differs but their synonym may match. Hence, synonym based matching approach is employed where the string to

be evaluated is compared with the synonym of the other string. If the synonym matches they are identified as duplicates.

Similarity between numeric fields is evaluated based on approximation. Approximation is sufficient to detect similarities in application like catalog integration. Domain knowledge on the numeric fields is used to determine the level of approximation. For example, in electronic goods database, a price difference of 1000 rupees between two similar products could be neglected. A record has multiple fields and each field has varying importance in contributing to record matching. Weights are assigned to various fields based on their importance in evaluating the similarity metric. Based on the value distribution of the field, suitable weight is assigned to the fields in order to specify the importance of the field in the record matching. Attribute entropy is an appropriate measure on structuredness of data distribution and weights are assigned to various fields based on their entropy. Higher weight is assigned on fields with less entropy value, since lower entropy indicates more number of duplicates in the value distribution of the field.

Matching string fields: Matching of string fields is carried out by comparing the characters and their sequence in the strings. If the two string fields are not matching exactly, their synonyms are compared for matching. For example, a product named computer could also be stored as personal computer or data processor in different data sources. Such string values cannot be matched using exact string matching methods. If exact string matching fails to match the string field values of the two records, their synonyms can be considered for matching. It is done by passing two strings to the WordNet ontology. WordNet ontology is a lexical database and it is used to retrieve synonyms of the string fields to be compared. It groups English words into sets of synonyms called synsets, provides short, general definitions and records the various semantic relations between these synonym sets. The purpose is 2 fold: to produce a combination of dictionary and thesaurus that is more intuitively usable and to support automatic text analysis and artificial intelligence applications. The synonymous words for a string value in one record are retrieved from WordNet. The synonyms are checked one by one to find if it matches the string value in the other record. If the string value matches with the synonyms from the WordNet, the two string values are identified as duplicates.

Matching numeric fields: For more efficient identification of duplicates, approximation approach is employed to evaluate the similarity between numerical values. A pre-defined threshold is fixed for the numeric field values in prior based on the domain knowledge. If the difference between the values is less than the pre-defined threshold, the values are treated as similar.

Weight assignment: Different fields may have different importance in identifying similar records. This importance is data dependent. Attribute entropy is a good measure of attribute importance that is calculated based on value distribution of the attributes (Yao *et al*., 1999). Regularities and structuredness are characterized by small entropy values whereas randomness is characterized by large entropy values. In other words, if an attribute has smaller domain, most of its data values occur multiple times in the given set of records. Attributes with duplicate values in several records will have smaller entropy value. Such attributes are important in identifying duplicate records. Higher weights are to be assigned for attributes with lower entropy. Entropy is inversely proportional to the number of duplicates. Normalized entropy is used to classify the attributes as important or non-important. Higher the value of normalized entropy, higher is the importance of the attribute in identifying the duplicate records:

$$H_N(X) = 1-\frac{H(x)}{\log_2(m)} \quad (3)$$

where, m is the number of records in the database. For primary key field, $H(X) = \log_2(m)$ and normalized entropy $H_N(x) = 0$ and hence key fields are less important in identifying duplicate records. Weights are assigned to a component to indicate the importance of a field under the condition that the sum of all component weights is equal to 1 because the similarity metric between two duplicate records should be close to 1.

Weight calculation: Assume that entropy for the attributes X_1, X_2, X_3,..., X_n is given by $H(X_1)$, $H(X_2)$, $H(X_3)$...$H(X_n)$. Then, weight assigned to an attribute X_j is calculated using Eq. 4:

$$W(X_j) = \frac{H_N(X_j)}{\sum_{i=1}^{n} H_N(X_i)} \quad (4)$$

where, $H_N(X_j)$ is the normalized entropy of the attribute X_j.

Similarity metric evaluation: To evaluate the similarity metric between two records, a similarity vector V is determined by comparing the values of different fields of the records. For each pair of records (r_1, r_2), a similarity vector $V = (v_1, v_2, \ldots v_n)$ where vi represents the ith field similarity between the records r_1 and r_2 is evaluated.

$v_i = 1$ if the ith field of the records are similar and $v_i = 0$ otherwise. Similarity metric between records r_j and r_k is defined as:

$$sim(r_j, r_k) = \sum_{i=1}^{n} (w_i \times v_i) \quad (5)$$

Where:

$$\sum_{i=1}^{n} w_i = 1$$

v_i = The similarity vector
w_i = The weight for the ith similarity component which represents the importance of the ith field

The similarity metric sim (r_j, r_k) between records r_j and r_k will be in the range [0, 1] according to the above definition.

Duplicate record elimination: A threshold value is fixed to classify records as duplicates and non-duplicates. If the similarity metric of the pair under consideration is above the similarity threshold then the records are treated as identical. Any one of the duplicate records can be remove from the data source in order to maintain consistency and save storage space. The step by step procedure to detect duplicate record pairs is described in SDRD algorithm:

SDRD algorithm:
Input: Integrated data source with n fields (f1, f_2…f_n) and m records (r1, r2…r_m).
Output: Data source D without duplicates.
Procedure:
[Weight assignment to various fields in the record]
for i = 1, 2…n
Compute Normalized entropy $H_N(f_i)$ every field f_i in D.
$H_N(f_i) = 1\text{-}H(f_i)/\log_2(m)$
for i = 1, 2…n
$W(f_i) = HN(f_i)/\sum_{i=1}^{n} H_N(f_i)$
[Compute similarity vector V (v_1, v_2…v_n)]
for every pair of records (r_j, r_k) in D where j = 1, 2,..., m-1 and k = j+1, j+2,..., m
for i = 1, 2,…, n
if (f_i is a string field)
if (StringCompare(v_jf_i,v_kf_i) = = 0)
$v_i = 1$
else
if (SynonymCompare (v_jf_i, v_kf_i) = = 0)
$v_i = 1$
else
$v_i = 0$
else if (f_i is a numerical field)
if (Approximation (v_jf_i, v_kf_i) = = 0)
$v_i = 1$
else $v_i = 0$
[Evaluate simialrity metric]
$sim(r_j, r_k) = \sum_{i=1}^{n} W(f_i) \times V_i$
[Detect duplicate records]
If sim (r_j, r_k) > = $, remove one of the duplicates r_k from D.
Exit

The time complexity of the proposed algorithm varies exponentially with the number of records in the data source, i.e, $O(m^2)$ where m is the number of records.

EXPERIMENTS

Experimental setup: Experiments were carried out on a 2.16 GHz processors with a 2 GB RAM with Windows XP Operating System. The implementation of this project is done using java. WordNet ontology is used for extracting the semantics of the string fields under consideration. JAWS is a Java API for WordNet searching that provides Java applications with the ability to retrieve data from the WordNet database. It is a simple and fast API that is compatible with many versions of the WordNet database files and can be used with Java.

Dataset: The product catalog for electronic goods were collected from 20 web sites and consolidated as a dataset with 150 entities. On average, the maximum number of duplicates for an entity is 10. The data records are randomly duplicated and a dataset of 5000 records was created. The product catalog has four fields including product_ID, Product_Name, Manufacturer_Name and Price.

Experimental results: A record linkage tool called FRIL (Fine-grained records integration and linkage tool), provides a rich set of tools for comparing records (Jurczyk *et al*., 2008). Furthermore, FRIL provides a graphical user interface for configuring the comparison of records. FRIL can be configured to choose different field similarity metrics and different record comparison techniques. For testing the product catalog integration, FRIL is configured to use Q-grams (Gravano *et al*., 2001) for comparing Product_Name field, exact string matching for Manufacturer_Name and numeric approximation for Price field. Nested loop join method that performs pair wise comparison of records is used for record comparisons.

The proposed SDRD Method uses exact string matching for Manufacturer_Name and synonym based comparison for Product_Name field. Precision and recall are the two measures used to measure the accuracy of the results returned by the record matching algorithms. Precision is used to check whether the results returned are accurate and recall is used to check whether the results returned are complete. Precision and recall measured for the proposed approach and FRIL is shown in Fig. 1 and 2, respectively.

When precision of the results is concerned both FRIL and the proposed approach have very close values.

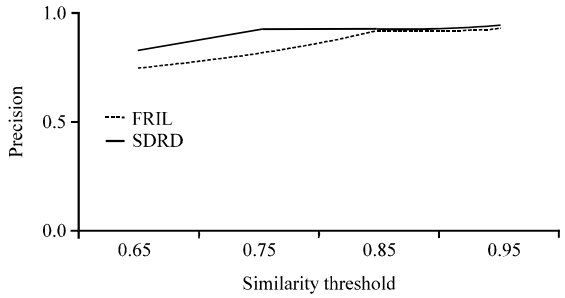

Fig. 1: Precision vs. similarity threshold

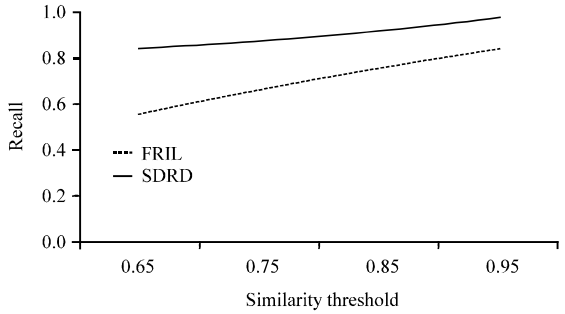

Fig. 2: Recall vs. similarity threshold

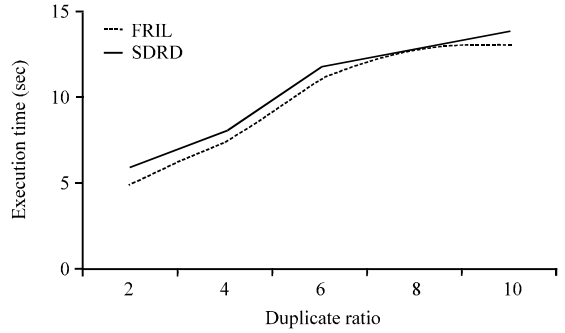

Fig. 3: Execution time vs. duplicate ratio

The proposed approach produces results with better recall compared to FRIL. Synonym based comparison determines additional matches, compared to string based field matching. As similarity threshold increases, the precision and recall of the results also increases because it decreases the number of false positives return by the record matching methods. On average, recall of the SDRD Method is improved by 20%. Figure 3 and 4 show the variation of execution time with the increase in data set size and for varying duplicate ratios.

Variation in execution time of the two approaches when the duplicate ratio is increased is shown in Fig. 3. The influence of large number of records on the execution time is shown in Fig. 4. For different duplicate ratio and varying number of records, the variation in execution time

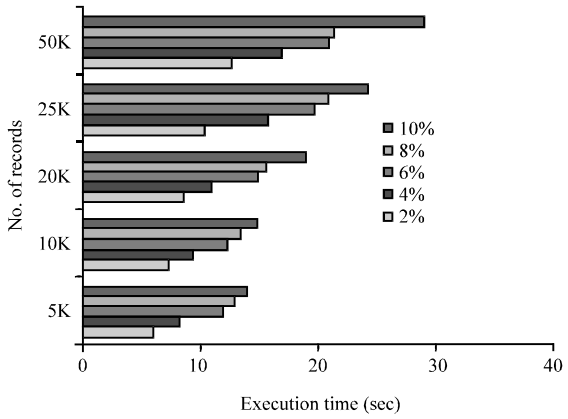

Fig. 4: Execution time vs. duplicate ratio for data sets of varying size

is linear. As the number of records and the duplicate ratio in the data source increases, execution time also increases.

CONCLUSION

The problem of identifying duplicates that is two or more records describing the same entity is of major concern when integrating heterogeneous data sources. Most of the earlier research is based on pre-defined matching rules and supervised method of detecting duplicates using trained data sets. A synonym based string field matching technique and entropy based weight assignment scheme are followed in the proposed approach which helps in detecting duplicate records that are missed by string matching methods. This duplicate record detection technique can be used to identify duplicates in data sets generated on the fly and formulate rules to detect database duplicates. Efficiency of the duplicate detection algorithm can be improved by reducing the number of record comparisons using techniques like blocking. In future the proposed approach may be combined with blocking to improve the efficiency of record comparisons.

REFERENCES

Cover, T.M. and J.A. Thomas, 1991. Elements of Information Theory. 99th Edn., John Willy and Sons Wiley-Interscience, USA., ISBN-13: 978-0471062592, Pages: 576.

Gravano, L., P.G. Ipeirotis, H.V. Jagadish, N. Koudas and S. Muthukrishnan *et al*., 2001. Using q-grams in a DBMS for approximate string processing. Data Eng. Bull., 24: 28-34.

Jurczyk, P., J.J. Lu, L. Xiong, J.D. Cragan and A. Correa, 2008. FRIL: A tool for comparative record linkage. Proceedings of AMIA Annual Symposium. November 8-12, 2008, Hilton Washington and Towers, Washington, DC.

Papadimitriou, P., P. Tsaparas, A. Fuxman and L. Getoor, 2012. TACI: Taxonomy-aware catalog integration. Trans. Knowl. Data Eng., 25: 1643-1655.

Su, W., J. Wang and F.H. Lochovsky, 2010. Record matching over query results from multiple web databases. Trans. Knowl. Data Eng., 22: 578-589.

Yao, Y.Y., 2003. Information-Theoretic Measures for Knowledge Discovery and Data Mining. Springer, Berlin, pp: 115-136.

Yao, Y.Y., S.K.M. Wong and C.J. Butz, 1999. On information-theoretic measures of attribute importance. LNCS Springer, 1574: 133-137.