Automated Solution for Normalization of Duplicate Records from Multiple Data Sources

K. Jaya Sri
M. Tech,
Department of CSE,
Shri Vishnu Engineering College for Women (A),
Vishnupur, Bhimavaram,
West Godavari District, Andhra Pradesh.

K. Ramachandra Rao
Ph.d, Associate Professor,
Department of CSE,
Shri Vishnu Engineering College for Women (A),
Vishnupur, Bhimavaram,
West Godavari District, Andhra Pradesh.

Abstract— There has been an exponential growth of data in the last decade both in public and private domain. The main aim of this project is to identify the duplicate records which represent the same real world entity by using a mechanism which does not require any training data. An unsupervised method is used where no manual labeling is required. Detecting data sources records that are approximate duplicates is an important task. Query and data from multiple data sources will result with duplicates. When information is retrieved from different data sources duplicates occur due to various format specifications. A data sources having unintentional duplication of records created from the millions of data from other sources can hardly be avoided. Data sources may contain duplicate records that represent the same real world entity because of data entry errors, abbreviations, detailed schemas of records from multiple data sources. Supervised methods are the current techniques used for duplication detection, which requires trained data. These methods are not applicable for the real time data source scenario, where the records to match are query results dynamically generated in online. I present a Dynamic Duplicate Detection, for a given query the algorithm can effectively identify duplicates from the query result records of multiple data sources. In the algorithm proposed, I start from the non-duplicate set and use a weighted component similarity summing classifier and an OSVM classifier, to iteratively identify duplicates in the query results from data sources. Additional to these two classifiers which are used in Unsupervised Duplicate Detection algorithm, a third classifier called Blocking Classifier is used which helps in detecting the duplicate records. Various experiments are conducted on a data set to verify the effectiveness of the algorithm in detecting the duplicate records.

Index Terms— Multiple Data Sources, Dynamic Duplication,

1. Introduction

Today more and more data sources generate web pages in response to user queries that are available on the web. These dynamic web data sources are estimated to have a much larger amount of structured information and also have a faster growth rate when compared to the static web. Most of the web data sources are accessible through query interface where user submits their queries. Once a query is received, the web server will retrieve the results from the data source and then return to the user. In the web data sources the records to match are based on the query given since they can be obtained through online queries. With the exponential growth of the web pages and end users demand for optimal search results, data mining techniques are been developing vastly to clear the process of understanding the data as well as pre-processing and data preparation. When dealing with large amount of data, it is important that there must be a well-defined and tested mechanism to filter out duplicate results. This keeps the end results relevant to the queries.

Duplicate records exist in the query results of many Web data sources, especially when the duplicates are defined based only on some of the fields in a record. Using exact matching technique as part of preprocessing, records that are exactly the same in all relevant matching fields can be merged. The techniques that deal with duplicate detection can be broadly classified as those requiring training data (supervised learning method) and those that can function without a predefined training data (unsupervised learning method. As part of this project, un-supervised techniques are been explored and then a mechanism is proposed where the function can be done with minimal supervision.

2. Problem Definition

The end user has no control over the results returned by the data sources, nor do they guarantee that there will be no duplicates from the query result. The problem of duplicate records existing in a query result referring to the same real-world entity can occur when search engine uses multiple web data sources. The focus is mainly on data sources from the same domain, i.e., Web data sources that provide the same type of records in response to user queries. Consider that there are p records in a data source A and there are q records in a data source B, with each record having a set of fields or attributes. Each of the q records in the data source B can potentially be a duplicate of each of the p records in the data source A. The goal of duplicate detection is to determine the matching status, i.e., duplicate or non-duplicate, of these p * q record pairs.

The problem now is that the search needs to identify records that refer to the same entity and display a unique set. If only exact matching algorithm is used, it wrongly identifies all the four records from are different and not unique. It was wrongly written as unique instead of not unique. The main work of this project is to develop a mechanism that can identify duplicate record pairs from the query results that refer to the same real-world entity using an unsupervised algorithm.

3. Proposed System

Unsupervised Duplicate Detection (UDD) algorithm is used to detect duplicate records. The project builds up on the idea of UDD and is an attempt to enhance the algorithm by introducing an efficient classifier to improve the accuracy rate. UDD assumed that there will be no duplicates within a data source and only considered the result set from multiple data sources for potential duplicates. The introduction of the blocking classifier helps in identifying the duplicates that are present within the data source. In the proposed system the plan was to develop an algorithm that uses three classifiers for detecting duplicate records.

The first classifier is called the Weighted Component Similarity Summing (WCSS) Classifier where the field calculations are done and duplicates are identified without any training. The idea for this classifier is to calculate the similarity between pair of records by doing a field to field comparison and then weight can be calculated. The non-duplicate records identified by the first classifier serves as input to the second classifier .The Support Vector Machine (SVM) Classifier makes use of the duplicates and non-duplicates identified from the WCSS classifiers as training dataset. The SVM classifier then uses the training data and processes each record to identify a record as being a duplicate or unique. Dynamic duplicate Detection develops a third classifier called the Blocking Classifier and integrating this with the current UDD process to better identify duplicates and enhance the process. The other major enhancement would be comparison of similarity functions that can help in identifying duplicates. An ideal system would be the one that presents the most relevant results from multiple web data sources. This project was an attempt to explore using alternative methodologies for record duplicate detection.

By developing additional classifiers, an unsupervised learning system becomes more suitable for dynamic duplicate detection. They are useful in finding the duplicates from the queries given dynamically. To measure the effectiveness of the proposed system, results of experiments were compared with the original UDD system.

4. Implementation of UDD algorithm

Input : Potential duplicate vector set P Non-duplicate vector set N

Output: Duplicate Vector set D

C1: a classification algorithm with adjustable parameters W that identifies duplicate vector pairs from P

C2: a supervised classifier

Algorithm step:-

```
step:1    D=Ø
step:2    Set the parameters W of C1 according to N
step:3    Use C1 to get a set of duplicate vector pairs d1 from P
step:4    Use C1 to get a set duplicate vector pairs f from N
```

```
step:5    P=P – d1
step:6    While |d1| ≠ 0
step:7    N'=N-f
step:8     D=D+d1+f
step:9    Train C2 using D and N'
step:10    Classify p using C2 and get a set of newly identified duplicate vector pairs d2
step:11   P=P - d2
step:12   D=D+d2
step:13   Adjust the parameters W of C1 according to N' and D
step:14   Use C1 to get a new set of duplicate vector pairs d1 from P
step:15   Use C1 to get a new set of duplicate vector pairs f from N
step:16    N=N' Return D
```

Initially, a classifier is trained using vector set N and the learned classifier along with another co-operating classifier works on vector set P. the records that are retrieved from the web database, first it will assign weights to the fields and identify similar and dissimilar records iteratively.

5. Architecture of the System

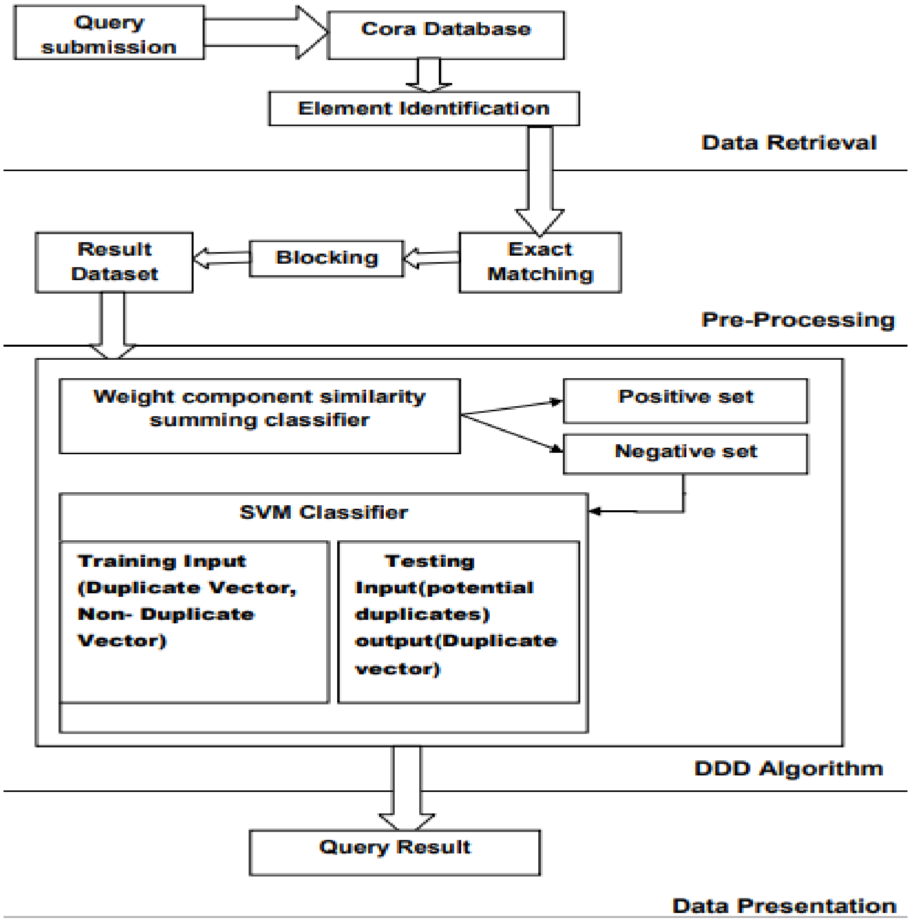

Figure: - Architecture of the system

The architecture of the system can be explained in detail with the various steps involved in the process. The architecture gives a process of how the system works on the dataset and identifies the duplicate records for the query given by the user. The below figure shows the system architecture of the proposed work. One of the major component of the system is the module that has the Dynamic Duplication Detection algorithm.

Developing our algorithm that can train itself and help in identifying duplicates is the main goal of the project. This algorithm consists of a component that calculates the similarity vectors of the selected dataset, assigning weights to the selected vectors and finally using the support vector machine (SVM) to classify the data. The potential duplicates are identified and the records are given without any duplication. The actual duplicates are eliminated and then the output is send to the user.

6. Conclusion

This project concentrated on the development of an Unsupervised Duplicate Detection algorithm that can serve as foundation for developing applications that use Web data sources. The above result tells that by using an additional blocking classifier can result in higher accuracy. With exponential growth of data, duplicate detection is an important problem that needs more attention. Using a Dynamic Duplicate Detection algorithm that learns to identify duplicate records has some advantages over offline/supervised learning methods. Although the focus of the Dynamic Duplicate Detection application in the project was limited to Cora dataset, the same principles can be used broadly to other domains. When compared to traditional data sources, Web-based retrieval system in which records to match are greatly query dependent, a pre-trained approach is not appropriate as the set of records in response to a query is a biased subset of the full data set. Dynamic Duplicate Detection algorithm which is an unsupervised, online approach for detecting duplicates is a suitable solution, when query results are fetched from multiple Web data sources. The core of Dynamic Duplicate Detection algorithm relies on using WCSS and SVM classifiers to assign weights and classify data. This project is a step forward in enhancing the UDD algorithm by adding an additional classifier.

Experimental results demonstrate blocking classifier were able to limit the number of record comparisons that take place thereby improving accuracy. 58 This was done by effectively grouping source data using similarity metrics. One observation was that using blocking produced almost similar results, which leads us to conclude that any hash function can be used in a blocking algorithm. A comparison of experimental results with the standard Dynamic Duplicate Detection algorithm showed blocking classifier was able to limit the number of false positives. Although the experiments were limited, this shows that UDD combined with blocking is comparable fast to other approaches for duplicate detection. Most of the current duplicate detection systems focus on two aspects, one using machine learning based algorithms to speed up the process of identifying duplicates and second, developing knowledge based approach for matching pairs of records. An interesting direction for future research is to develop techniques that combine these two approaches. This is used for development of robust and scalable solutions. More research is needed in the area of data cleaning and information quality in general and in the area of duplicate record detection in particular.

Reference

[1] R.Ananthakrishna, S. Chaudhuri, and V. Ganti, “Eliminating Fuzzy Duplicates in Data Warehouses,” Proc. 28th International Conference Very Large Data Bases, 2002, pp. 586-597.

[2] Amy J C Trappey , Charles V . Trappey, Fu – ChiangHsu and David W Hsiao , “ A Fuzzy ontological knowledge Document Clustering Method”, IEEE Transaction on Systems, Man, Cybernetics, June 2009, Vol 39 No. 3.

[3] Baodong LI, Yongquan DONG, Yongxin ZHANG and DonglanLIU, ” Duplicate Record Detection Based on Unsupervised Learning Method”, Journal of Computational Information Systems, December 2011, Vol. 7, No. 16, pp. 5891-5899.

[4] Bolla Anil Kumar, Satya P Kumar and Somayajula, “Hide the Duplicate Web Pages”, International Journal of Computer Science and Technology, September 2011, Vol. 2, No. 3, pp. 438-440.

[5] R. Baxter, P. Christen, and T. Churches, “A Comparison of Fast Blocking Methods for Record Linkage, ” Proceedings Knowledge Discovery on Data Workshop Data Cleaning, Record Linkage, and Object Consolidation, 2003 , pp. 25-27

[6] R. Baxter, Lifang Gu ,”Adaptive Filtering for Efficient Record Linkage”, SIAM International Conference on Data Mining, 2004, pp.477-481

[7] M.Bilenko and R.J. Mooney, “Adaptive Duplicate Detection Using Learnable String Similarity Measures,” Proceedings ACM SIGKDD conference on Knowledge Discovery and Data mining, 2003, pp. 39-48.

[8] Cai Bo, Zhang Feng Li and Wang Can, ” Research on Chunking Algorithms of Data De-duplication”, American Journal of Engineering and Technology Research, 2011, Vol. 11, No. 9, pp. 1353-1358.

[9] P.Christen, “Automatic Record Linkage Using Seeded Nearest Neighbour and Support Vector Machine Classification,” Proceedings ACM SIGKDD conference on Knowledge Discovery and Data mining, 2008, pp. 151-159.

[10] P.Christen and K. Goiser, “Quality and Complexity Measures for Data Linkage and Deduplication”, Springer, 2007, vol. 43, pp. 127-151.

[11] S.R. Motwani, “Robust and Efficient Fuzzy Match for Online Data Cleaning,” Proceedings Knowledge Discovery and Data mining 2003, pp. 313-324.

[12] S. Chaudhuri, V. Ganti, and R. Motwani, “Robust Identification of Fuzzy Duplicates,” Proc. 21st IEEE International Conference on Data Engineering, 2005, pp. 865- 876.

[13] DebabrataDey, Member, IEEE, Vijay S. Mookerjee, and Dengpan Liu, “Efficient Techniques for Online Record Linkage”, IEEE Transactions on Data Engineering, March-2011, Vol. 23, No. 3, pp. 373-387.

[14] Diego Zardetto, Monica Scannapieco and TizianaCatarci, “Efficient Automated Object Matching”, International Council for Open and Distance Education World Conference, March 2010, pp. 757-768.

[15] V.S. Verykios. “Duplicate Record Detection: A Survey”, IEEE Transaction Knowledge and Data Engineering, 2007, pp. 1-16.

[16] Haibin Cheng, Pang-Ning Tan, Member, IEEE, and Rong Jin, “Efficient Algorithm for Localized Support Vector Machine,” IEEE Transaction Knowledge and Data Engineering, April 2010, vol. 22, no 4

[17] “PEBL: Web Page Classification without Negative Examples,” IEEE Transaction on Knowledge and Data Engineering, Jan. 2004, vol. 16, no. 1, pp. 70-81.

[18] Ho Min Jung_, Sang Yong Park, Jeong Gun Lee, Young Woong Ko, “Efficient Data deduplication System Considering File Modification Pattern,” International Journal of Security and Its Applications.April, 2012 Vol. 6 No. 2.