Effective Compatibility and Reduction of Data for Bigdata Applications

Vikas S Thimmaraju S N

Abstract- ***The system identifies a duplicate record from the database using the machine learning method. We must pass unstructured data. Data are prepared using any natural language processing technique such as text similarity. This prepared data is then fed into the latest machine learning method called Random Forest. After this data collection, using these files, the target file is compared to the source file. We make input and output files. This is carried out until accurate efficiency is generated.***

Keyword: We Data processing fficiency enerated

I. INTRODUCTION

Record linkage is the recognition method when two documents of the database refer to the same entity. Record connection is a basic issue when various sources of data are integrated. For instance, one data source may refer to "International Animal Productions" with headquarters in "Los Angeles, CA," and another may refer to "Intl" as the same business. The "Hollywood" animal. The general issue of two references to the same entity are commonly recognized and versions of the issue include ' deduplication, ' ' object identification ' and
' co-reference resolution '.

Union Switch and Signal	2022 Hampton Ave	Manufacturing
JPM	115 Main St	Manufacturing
McDonald's	Corner of 5th and Main	Food Retail
Joint Pipe Manufacturers	115 Main Street	Plumbing Manufacturer
Union Sign	300 Hampton Ave	Signage
McDonald's Restaurant	532 West Main St.	Restaurant

Figure 1. Matching Records in Two Tables

A critical aspect of matching two documents is assessing how well each field matches (i.e. characteristics). Record connection schemes usually use similarity metrics comparing pairs of field values, such as two addresses, and returning a measure of their resemblance. In view of similarity assessments at the field level, an general judgement at the record level is produced.
Researchers in machine learning have taken a distinct approach to recording the connection. They have created systems using advanced record-level decision-making techniques such as decision trees, supporting vector machines, and unsupervised statistical methods.

Revised Manuscript Received on October 15, 2019.
Vikas S, VTU PG Centre Mysuru, Karnataka, India
vikas.smg@gmail.com
Dr.Thimmaraju S N, VTU PG Centre Mysuru, Karnataka, India
Thimmaraju_sn@yahoo.com

However, they depended mainly on easy generic techniques, such as TF-IDF and other string-similarity metrics, for field-level similarity assessments. These tend to be generic and homogeneous for each sector as opposed to expert systems.
Overview Of Our Approach
The record linkage strategy outlined here assumes that in two database tables, A and B, we are linking data so that in each table there are associated characteristics, each table includes components of the same sort. There are extra complexities in many apps; for example, one table may have two characteristics, such as "first name" and "last name," and the other table may have "complete name" characteristics. In general, these complexities can be addressed in a pre-processing stage (e.g., concatenating "first name" and "last name"), which is why our strategy is relevant in a broad variety of environments.
Our method of connecting records has several stages. First, in each record, we parse each cell into a collection of tokens, where each token is a single word, number, or symbol. We also optionally label the tokens with a semantic category (e.g., parsing a full name in the first name, optional middle initial, and last name) and optionally apply a set of standardization operators to standardize the tokens.
Second, to recognize pairs of documents that have the ability to match, we use a blocking algorithm. This eliminates the need for a complete cross-product evaluation. We use a reverse index in our application to define possibly matching pairs, comparable to the methodology outlined in[11] that takes many of the transformations into consideration.
Next, each pair of candidate documents (Aj, Bk) are taken and compared field by field. I.e., we assess each pair of values (Aji, Bki) using an acquired Fi metric distance.
We current Dedoop (Hadoop duplication), an MapReduce (MR)-based ER framework. The MR programming model is well suited for ER as it is possible to execute parallel pair-wise similarity computation. Using a cloud infrastructure considerably speeds up ER programs and therefore has several benefits. First, ER parameter manual tuning is facilitated as ER outcomes can be produced and assessed rapidly. Second, the decreased execution times for big data sets accelerate popular data management procedures, e.g. data warehouse ETL programs.
The highlights of Dedoop are as follows:
• In a web browser, Dedoop allows users to readily specify sophisticated ER workflows. This allows users to choose from a wealthy set of popular ER parts (e.g. blocking methods, similarity features, etc.) including machine learning to build match classifiers automatically.
• Dedoop automatically transforms the definition of the workflow into an executable workflow for MapReduce. It is possible to visualize the ER outcomes and the workload of all cluster nodes afterwards.

• In conjunction with its blocking techniques, Dedoop offers several load balancing strategies to obtain balanced workloads across all cluster nodes employed. It can also prevent unnecessary entity pair comparisons resulting from various blocking keys being used.

II. IMPLEMENTATION DETAILS

The proposed Method can be put into the following modules

- Corpus Creation
- Pre-Processing
- UPC Exact Match
- Manufacturer Name, Manufacturer Part No. and Brand Name Exact Match
- Synonym Handler
- Feature Selection
- Random Forest

Corpus Creation

The system primarily acquires Source Data from the File given by the User and uploads Target Data from Database. This Data will contain information about individual entities. Each Entity will have set of attributes such as Item ID, Product Name, UPC, Manufacturer Name, Manufacturer Part No, Brand Name, Product Description and Taxonomy Classification. Based on these attributes the system identifies whether two records are identical or not. This is achieved by going through a series of steps including Machine Learning and Natural Language Processing Techniques.

Pre-Processing

This stage is the essential phase of the system. Only after this stage the further process can be completed. We need to prepare the data, before the records can be used for the process of identifying identical records by using the Machine Learning Concept. In this stage there are several processes taking place. They are Abbreviation and Synonym Handler, Tokenize Handler, StopWords Handler and Cross Join. In Abbreviation and Synonym Handler systemconsiders Brand Name and Product Description attribute of an entity. Product Description attributes is a sentence describing the entity. But since the Source Data is acquired from the User the Product Description will contain few Abbreviated words which needs to be handled so that it is easy to find the identical records. System replaces every Abbreviated words in the Product Description with their expansions.

StopWords are words that are automatically omitted from a computer-generated concordance or index. Computing on these StopWords are waste of resource so to reduce the computation we remove these StopWords from the Product Description. To remove these StopWords we need to break the sentences into collection of words. To achieve that Tokenize Handler is used. Tokenize Handler breaks the sentences into collection of Words. We are using Regex Tokenizer so along with breaking the sentences into words it will remove unwanted special characters. After the sentence is split into words we intend the system to find the StopWords in that collection of Words and remove those words which are not required for Computation. This is done by StopWords Handler.

After the Stop Words are removed Cross Join the Source Data and the Target Data. Since the further process of the system is complex Cross Join is done. Cross Join joins each record from Source Data with Every record on the Target Data and stores it into a new Dataset. This new Dataset contains both the records from Source as well as Target therefore whatever the data is required system doesn't need to create instances for both Source and Target Data. The Data is now prepared and now the system proceeds with further process.

UPC Exact Match

First step in the process of identifying the identical records is UPC Exact Match. UPC attribute of an entity is a combination of 12, 13 or 16 digit numbers. The system considers UPC attribute of an entity from Source Data and finds the exact same UPC value which matches the UPC of an entity from the Target Data. After finding the matched UPC records it will be displayed as the output of UPC Exact Match module.

Manufacturer Name, Manufacturer Part No and Brand Name Exact Match

Manufacturer Name, Manufacturer Part Number and Brand Name attributes of an entity are in the form of string. The system considers Manufacturer Name, Manufacturer Part Number and Brand Name from the Source Data and finds the exact same Manufacturer Name, Manufacturer Part no. and Brand Name values of an entity from Target Data. After finding the matched records it will be displayed as the output of this module.

Synonym Handler

In this step, the Synonym (alternative word) is handled by replacing the words from the source data with its equivalent data which is given in a list of synonyms.

Feature Selection

In this stage, the System there are processes that are done to identify identical records. They are Feature Selection, String Similarity Calculation, Vector Representation and Random Forest. System uses the Pre-Processed Data from the previous module. Random Forest is a Machine Learning Technique which finds identical records based on the features that is selected. Features are nothing but attributes of an entity. In this stage we intend the system to select which features to choose for identifying identical records. Therefore, we intend system to select attributes such as Product Description, Manufacturer Name and Manufacturer Part Number from the processed data.

Since Random Forest cannot predict the identical records which are in String Format so features that has been selected should be represented as scores. To achieve that String Similarity Calculation is done. String Similarity Calculation can be achieved by using String Similarity Metrics from the Natural Language Process Techniques. String Similarity Metrics like Cosine Similarity, Jaccard Similarity and Sorensen-Dice Similarity are used to calculate the Similarity.

Random Forest accepts the data as input in a specific format. That format is called libsvm (Library for Support Vector Machines). System represents the scores got from the String Similarity Metrics into Vectors. This is done by the Vector Representation process. After this process Sorted Neighborhood method is implemented because after Cross Join the amount of records will be very large in number. To reduce the number of computation cycle we use the Sorted Neighborhood Method. In this Method, every set of records are sorted in descending order based on the String Similarity Scores and top 4 records are selected. These top 4 records are then supplied as input to the Random Forest.. Using the same procedure Random Forest creates decision trees using the records given as the input and based on the votes of the decision tree records that are identical are predicted. The predicted records are then displayed as the output of this module.

III. EXPECTED RESULTS

- This research will assist manage big volumes of information and use the Apache Spark framework to effectively manage big volumes of information. The ability to generate a deduplicated information set based on matching outcome, approaches to string resemblance and method of machine learning. The enrichment of product information provides clients and vendors satisfaction.
- By Appling above method we found the Accuracyfor the First Time 35 % for the total Prediction of 59 off 100 but the correct predictions are 35.
- After applying normalization methods we found the Accuracy by 53% fortotal Prediction of 71 off 100 but the correct predictions are 53. Total Train set: 3902Test Set: 400.

Fig2: Displaying Final Result

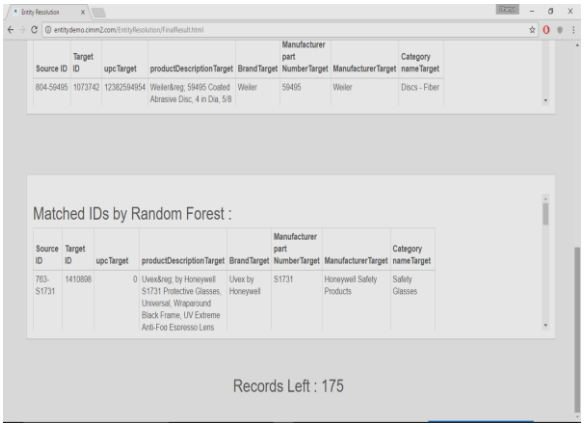

Fig3: Displaying Final Result with the Count of Records Unmatched

REFERENCES

1. Pucktada Treeratpituk, C.Lee Giles Information Sciences and Technology Pennslyvania State University, *JCDL'09,* June 15–19, 2009, Austin, Texas, USA. Copyright 2009 ACM 978-1-60558- 322-8/09/06
2. R. Bekkerman and A. McCallum. Disambiguating Web appearances of people in a social network. Proc of Int'l World Wide Web Conf (WWW), 2005.
3. O. Benjelloun, H. Garcia-Molina, H. Kawai, and T. Larson. D-swoosh: A family of algorithms for generic, distributed entity resolution. Proc of the 27th Int'l Conf on Distributed Computing Systems, 2007.
4. M. Bilenko, B. Kamath, and R. Mooney. Adaptive blocking: Learning to scale up record linkage. IEEE Steven N. Minton and Claude Nanjo, Fetch Technologies 2041 Rosecrans Ave., Suite 245 El Segundo, CA 90245.
5. M. Bilenko and R. J. Mooney. Adaptive duplicatedetection using learnable string similarity measures. In Proceedings of ACM SIGKDD-03, pages 39–48,Washington DC, 2003.
6. P. Christen, T. Churches, and J. X. Zhu. Probabilistic Name And address cleaning and standardization. In Proceedings of the Australasian Data Mining Workshop,2002.
7. Dedoop. http://dbs.uni-leipzig.de/dedoop.
8. Baxter, Christen, and Churches. A Comparison of Fast Blocking Methods for Record Linkage. In Workshop Data Cleaning, Record Linkage, and Object Consolidation, pages 25–27.
9. Lars Kolb,Database Group,University of Leipzig "Dedoop: Efficient Deduplication with Hadoop".
10. Kolb, Thor, and Rahm. Load Balancing for MapReduce-based Entity Resolution. In ICDE, pages 618–629, 2012.

AUTHORS PROFILE

Mr.Vikas S, received M.Phil degree in Computer Science in the year 2009 and Master of computer Applicaions (MCA) in the year 2007 from Visvesvaraya Technological University and Bachelors Degree in Computer Science in the year 2004 from kuvempu University. He is currently working as Assistant Professor in the Department of MCA, Visvesvaraya Technological University, PG Center, Mysore, Karnataka, where he is involved in research and teaching activities. He is having 11 years of teaching experience and 02 years of Industrial experience. He is a Life member of India Society for Technical Education (LMISTE), Computer Society of India (CSI) and Doing Research work on the Area Big data Analytics.

Dr. Thimmaraju S N, he is presently a professor and heading the Department of Master of Computer Applications, Visvesvaraya Technological University, PG Center, Mysore, Karnataka, he has received his Ph.D degree from Visvesvaraya Technological University(VTU), Belgaum in the year 2010, M.E., degree in Computer Science and Engineering from University Visvesvaraya College of Engineering (UVCE), Bangalore in 2002 and Bachelors Degree in Computer Science and Engineering from PESCE, Mandya in the year 1999. He is involved in research and teaching activities. His major areas of research are Computer Networks, WSN's and Graph theory. He is having 17 years of teaching experience. He has published around 17 research papers which include International Journals, International Conferences and Notional Conferences.