A HYBRID INTELLIGENT SYSTEM TO IMPROVE DATA PREPROCESSING

Sohail Sarwar[1], Zia Ul Qayyum[2],Muhammad Safyan[1],Abdul Kaleem[1]

[1] Department of Computing Iqra University Islamabad,Pakistan

[2] National University of Computing and Emerging Sciences (FAST) Islamabad, Pakistan

sohail.sarwar@seecs.edu.pk, zia.qayyum@nu.edu.pk,m.safyan@seecs.edu.pk,abdulkaleem@yahoo.com

ABSTRACT. *Data preprocessing having a pivotal role in data mining ensures reduction in cost by catering inconsistent, incomplete and irrelevant data through data cleansing to assist knowledge workers in making effective decisions through knowledge extraction. Prevalent techniques are not much effective for having more manual effort, increased processing time, less accuracy percentage etc with constrained data volumes. In this research, a comprehensive, semi-automatic pre-processing framework based on hybrid of two machine learning techniques namely Conditional Random Fields (CRF) and Hidden Markov Model (HMM) is devised for data cleansing. Proposed framework is envisaged to be effective and flexible enough to manipulate data set of any size. A bucket of inconsistent dataset (comprising of customer's address directory) of Pakistan Telecommunication Company (PTCL) is used to conduct different experiments for training and validation of proposed approach. Small percentage of semi cleansed data (output of preprocessing) is passed to hybrid of HMM and CRF for learning and rest of the data is used for testing the model. Experiments depict superiority of higher average accuracy of 95.50% for proposed hybrid approach compared to CRF (84.5%) and HMM (88.6%) when applied in separately.*

1. INTRODUCTION

Mammoth masses of data are piling up every day at a great pace making it a challenging task to extract useful information (i.e. data mining) aiding in intelligent decision making through machine learning techniques. Effectiveness of these techniques in making right decisions greatly relies on quality of data instilled into data repositories (i.e. data warehouses) where data needs to be correct, consistent, less redundant and structured in a meaningful way [1,2,3]. Presence of these features in data can be ensured through incorporation of certain data preprocessing techniques (i.e. Data Cleansing) [4, 5]. Some tools provided by Oracle, Premier International, SQLPower, Informatica [6,7,8,9] are also available for preprocessing of data but they are expensive and not comprehensive enough to cater large data size and variety of cleansing situations (more suited to European standards of data expression) since these tools operate in a rule-based fashion i.e. if-then-else.

Keeping in view this pivotal role of data preprocessing, we present based semi-automatic model based on artificial intelligence techniques to perform data preprocessing with minimum time, human effort and improved accuracy level. This solution targets data preprocessing of a Telco's address directory named PTCL providing voice and data services across the Pakistan with approx 3200 telephone exchanges having 7 million customers [10]. In addition to asserted features, following issues associated with locale of Asia (esp. PTCL, Pakistan), where data logging is not standardized contrary to European locale, have been addressed:

- **No atomicity** in "91, 43, F-10/4 and ISLAMABAD" in values of address "HOUSE NO.91, ST-43, F-10/4, ISLAMABAD". These values should be placed into fields "HOUSE", "STREET", "SECTOR" and "CITY" of an address table respectively.
- **Non-Standard abbreviations** for "ISLAMABAD" in PTCL address directory, like "ISB", "IBD", "ISL" etc
- **Inconsistency** is, "H # 18, ST # 23, SECTOR I-9/4, ISLAMABAD" and "18, ST 23, I-9/4, ISLAMABAD" are both address of one physical location etc.
- **Incorrect and misspelled data** is stored, like "UMS" used instead of "LUMS".

Currently there is no effective mechanism adopted by PTCL that can overcome above problems in customer addresses. Each customer's address in whole directory is stored in single column of database table, without following any standard and pattern. So it appears to be quite challenging task to correct millions of prior records congregated over years. Manifold approach, with mutual comparison of probabilistic/statistical techniques, has been designed to address the above issues in order to cleanse the subject dataset of enormous customer base.

Proposed approach, as illustrated in Fig. 1, involves following steps:

- In **Pre-training / preprocessing** phase tokenization of addresses, generation and maintenance of data dictionary, identification of duplicate tokens, replacement of incorrect tokens, replacement of abbreviations, and assignment of tags to the tokens has been handled.
- In **Training phase** System is trained through supervised training using Hidden Markov Model (HMM), Conditional Random Fields (CRF) and combination of both alternatively[2, 5, 11, 12, 13]. Selection of record would be in random way. It would also provide facility to make training automated.
- In **Testing phase** model would be able to test System through HMM and CRF for rest of the data. The proposed model would **compare** the results of both techniques.

Comparison of these techniques helped to decide which one of these is better for data cleansing, segmenting and correction especially on addresses from Asian locale as well as European style. The intended data sets to be used for testing and validation are from Pakistan Telecommunications Limited (PTCL).

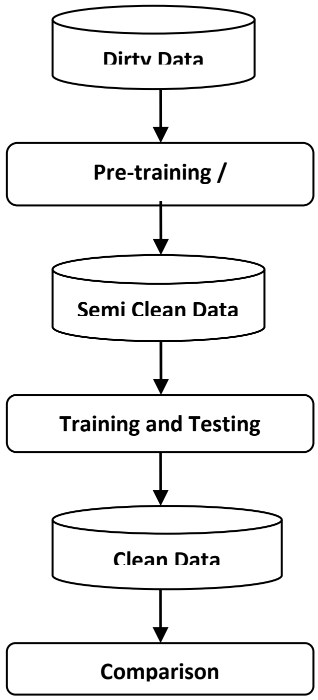

FIG. 1: PHASES OF DATA CLEANSING

ALLREC (e1,e2) in EMP
IF e1.name is similar to e2.name AND
e1.add=e2.add
THEN e1 matches e2

Rest of the paper is organized as follows: Section 2 covers the literature review of different data cleansing techniques, such as labeling sequences, probabilistic and non-probabilistic record matching techniques, Section 3 provides the details of proposed solution and, implementation of proposed solution and results are discussed in section 4 followed by conclusion and future work in section 5.

2. Literature Review

Labeling of a sequence [14, 15] can be considered as a set of classification tasks (independent from each other), one per each member of the sequence. In order to attain higher accuracy, the best way is to make the most appropriate (best) label for a given member depending upon the appropriate choices for adjacent member(s). It is done with the help of algorithms which are specialized in choosing the best set of labels (globally) for full sequence.

Inputs: I= (i1, …, in)
Labels: L = (l1, …, ln)
Typical goal: Given i, predict L

There are many models available for matching problem but these are limited for a particular type of data. Some of the models discussed in [16], mostly knowledge based, distance based, induction based as well as learning both supervised and unsupervised.

If an algorithm has capability to detect duplicate records in database; in several applications without any change in the algorithm than it is to be considering domain independent algorithm. The authors [17, 18] propose domain independent algorithms whose basic idea is "to apply algorithm for general-purpose field-matching".

In [16], an equational theory is discussed that uses some conditions for matching or identifying the domain equivalence. For example, if two employees in a table EMP have spelled their names nearly enough and addresses of those employees are the same then it may be concluded that both employees information is of one person. A declarative rule language is requiring for specifying such rules for using the equational theory model. Following example explains how one axiom of the equational theory is developed:

Supervised learning [19] is used in non-probabilistic way i.e. an algorithm based on machine learning used to generate rules for matching. When an algorithm has been selected then parameters are pruned, as a result a less complex matching rule generated. After development on a sample dataset of the improved matching rule, then it is used on the original large dataset. The CART algorithm is used in [19] produces linear combination of the parameters for data classification. It is a nearest neighbor algorithm's generalized form. The names and addresses parameters used with descriptors from a small sample of dataset, in order to build the matching rules.

The distance based matching techniques is all about finding the distance between two records which depends upon the weighted sum of the records. The weighted sum is calculated from the distance between weighted sums of the records' attribute values. In [20], limitation of probability models is addressed that is in case of missing accurate estimates and counts of probability parameters due to absence of manually matched training data. It uses simply distance based technique to overcome this problem.

Using probabilistic techniques, labeling sequences can be dividing into following components [17]: that is, what are the probabilities among states for an observation with conditions. Next is the identification of a procedure which finds efficiently best output labeled sequence from all possible candidates. There are two different probabilistic methods discussed: Hidden Markov Models (HMM) and Conditional Random Fields (CRF). Hidden Markov Model (HMM) is the directed graph, where as conditional random fields is an undirected graph.

An HMM is fully represented (mathematically) by two variable, x and y and two probability distributions A and B [2, 5]. Following is the explanation of the two variables:

- x represents the number of states given in the model.
- y represents number of distinct observation symbols
- In this case A represents the state transition probability distribution and A = aij (here 'aij' denotes the probability of transition from state 'i' to state 'j'.)
- B represents probability distribution of the observation symbol and B = bj(k). Here, the symbol set 'bj(k)' is the probability of emitting the 'k th' dictionary symbol in state 'j'

CRFs are categorized as probabilistic models for labeling sequence based on conditions that is attribute set of an observation. Technically these attributes called feature functions. CRFs are undirected graphical models [21]. When CRF models the conditional probability, there is a single joint probability distribution over the entire label sequence given in the observation sequence. This is in contrast to defining per-state distributions over the next states given in the current state [22].

The characteristics of the distribution over label sequences enable the CRF to model real-world data in which condition based probability can be determined. Conditions can be the sequence, type and nature of the observation sequence. Probability depends upon interacting features of the conditions. Also, the exponential nature of the distribution allows features of several different states to be exchanged / traded off against one another [22], weighing some states more in a sequence as considered higher in importance than other ones.

As the output sequence may be created using multiple paths with each path containing some value of probability. If there are 'x' states and 'y' represents length of the sequence, then

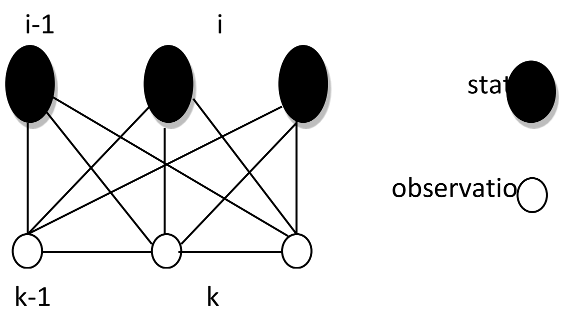

FIG. 3: CONDITIONAL RANDOM FIELDS AS AN UN-

there may be **O(yx)** possible paths that would generate the given sequence. The exponential complexity for finding the most probable path may be a bit high and it can be reduced to **O(kn2)** by the Viterbi Algorithm based on dynamic programming. Instead of summing up probabilities from different paths coming to same destination Viterbi picks up the best path and remembers it [5, 15]. Algorithm and pseudo code can be seen in [16].

Sorted neighborhood (SN) is one of the most famous approaches. The working goes like initially a blocking key '**K**' is defined for each of '**n**' entities. In general the blocking key is formed by concatenating the pre-fix of some attributes. In next step, sorting is performed on entities using this blocking key. Then a fixed size frame 'f' is applied over the records (already sorted) and in each step, all entities falling within the frame 'f' are compared. However, the range of distance would be f-1. [14].

3. Proposed Architecture

The proposed solution has two phases i.e. 1^{st} phase is pre-training cleansing and Tagging and 2^{nd} phase is of Training and Testing

3.1 Pre-Training Cleansing and Tagging (Phase-I)

This phase removes un-necessary words and symbols such as "no.","#","-" ,"," for not providing meaningful information but making the training cumbersome by adding noise. Moreover, anomalies of multiple abbreviations of same word like "ISB","ISL",|IBD" etc for "Islamabad" is addressed in addition to fixing incorrect spellings like "Islamabd" for "Islamabad" or "UMS" for "LUMS" followed by "tagging" of words. So we end up with semi-cleansed database having no unwanted words/delimiters and most of the words with correct spellings. These addresses are suitable input for training and testing phase carried out through AI techniques. Each process of this phase is briefly discussed in the following:

3.1.1 Distinct Words Dictionary

This process tokenizes all addresses on the basis of characters, digits, delimiters and strings and creates a Dictionary. Dictionary contains all the distinct tokens [i.e. words, numbers, delimiters, characters etc] with their frequencies. Dictionary is used to get the names of Area, streets, companies, Roads and buildings etc.

There is a large number of distinct tokens (words) extracted out of massive large data set. Linear search, especially in case of alphas, slowed overall processing and performance of the system in replacement, training and testing phase.

Therefore, a 3-dimensional data structure has been employed for search optimization where two dimensions of the structure have fix length that is dictionary [1] [1], each index representing one alphabet; the third dimension grows dynamically which contains singly linked list of distinct words. A typical representation of 3-D data dictionary is given in Fig 5.

3.1.2 Data Cleansing

This component of the system is used to assign a standard word to different variations of that word and eliminates the useless words from the records. Basic use of sorted neighborhood method is to find different variation of single word by creating keys. For identification of the different variation of a single value (word), Sorted Neighborhood is one of the best approaches. The brief description of Sorted Neighborhood used into model is:

- **Keys Creation:** A key is generated for every distinct word. There are two steps for key generation. In first step all vowels are eliminated and in second step repetition of consonants is eliminated.
- **Sort data:** On the basis of key, records are sorted and grouped.
- **Merge:** Here, all words having the same key are observed and if there was any incorrect word(s), then a correct alternate value (word) is assigned for replacement.

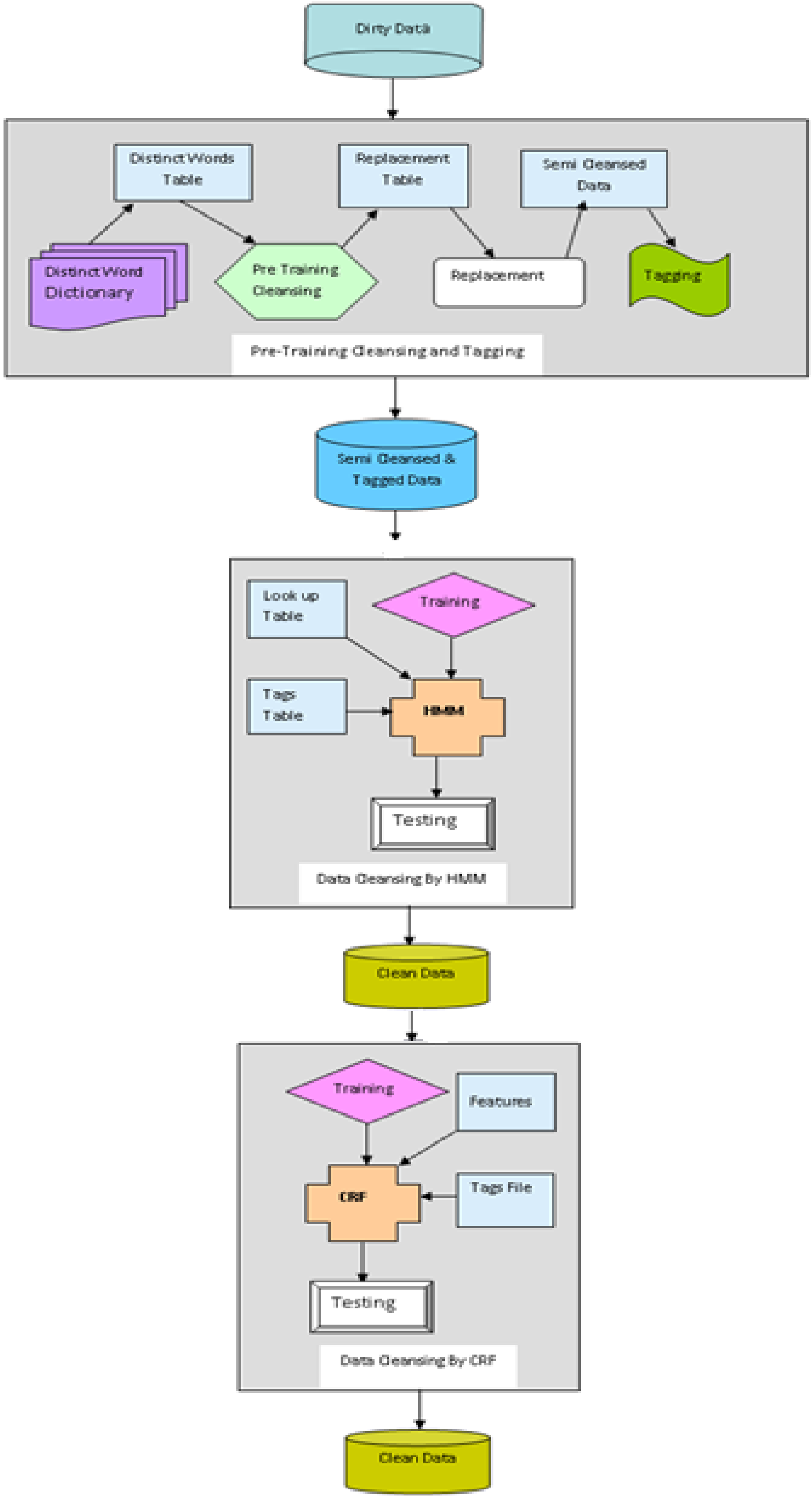

FIG. 4: PROPOSED ARCHITECTURE BASED ON HYBRID OF ML TECHNIQUES

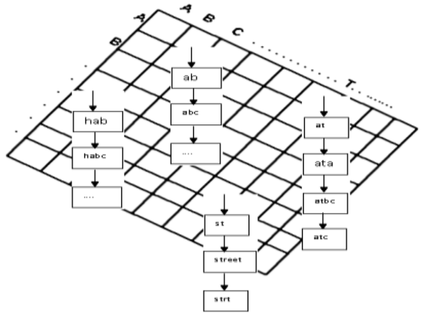

FIG. 5: REPRESENTATION OF DATA DICTIONARY

WORD	KEY
COLONY	CLNY
RAWLPINDI	RWLPND
MUHMMAD	MHMD
MOHMMAD	MHMD
RAWALPIND	RWLPND
CLONY	CLNY
CLNY	CLNY
MOHMAD	MHMD

FIG. 6: KEY GENERATION BY IGNORING VOWELS AND REPETITION OF CONSONANTS

3.1.3. Replacement

While using sorted neighborhood technique different tokens are shown to user on the basis of same key, which provides ease to correct them manually by identifying incorrect spellings in pre-training process. Once all incorrect distinct tokens are checked and correct alternate is assigned then words into database are automatically replaced by replacement process.

WORD	KEY	REPLACEMENT
RAWALPIND	RWLPND	RAWALPINDI
RAWLPINDI	RWLPND	RAWALPINDI
COLONY	CLNY	-
CLONY	CLNY	COLONY
CLNY	CLNY	COLONY
MUHMMAD	MHMD	MOHAMMAD
MOHMMAD	MHMD	MOHAMMAD
MOHMAD	MHMD	MOHAMMAD

FIG. 7: REPLACEMENT ASSIGNED TO NON-STANDARD VARIATIONS OF A WORD

All the non-standard variations of a word will replaced with standard assigned word into database by the end of this process.

3.1.4 Tagging

Tags are assigned to all the distinct replaced words into the dictionary according to their nature. It creates look-up tables used in training to identify tags for the tokens in the records.

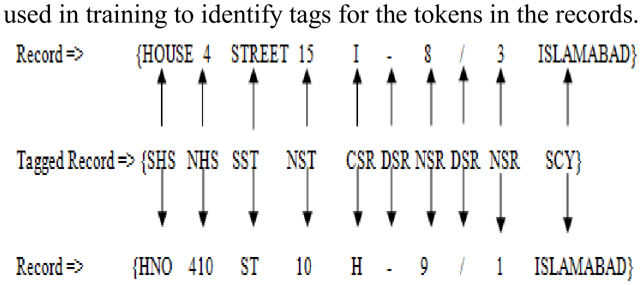

FIG. 8: TAGS ASSIGNED TO RECORDS

It is the most critical process. The accuracy of result is directly related to tagging; if a word is tagged wrongly then it would affect training of the machine learning techniques which would produce inaccurate result.

3.2 Machine Learning (Phase-II)

This phase consists of training and subsequent testing of machine learning models (HMM, CRF and alternate Hybrid of both) on data produced in phase-I that is asserted as training set and testing set respectively. Different sets of data buckets are used in training and testing phases. However, data for all three models remains the same in respective training and testing phases. Effectiveness of these techniques is evaluated based on degree of their elementization i.e. address is segmented into its atomic unit such that values are placed in their appropriate field (column) correctly.

4. Implementation and Evaluation

For implementation of HMM, MS Visual studio 2010 is used as IDE with MS Access as backend database in MS Windows 7 environment. Training of HMM is followed by testing where most of the steps are performed automatically or with less manual effort.

CRF code is taken from the website www.crfsharp.codeplex.com. Only one modification is made into code that is test/ output data is also stored into MS access database besides the notepad file for the comparison among HMM, CRF and their hybrid.

At first, unclean, un-segmented and incorrect 5000 addresses were taken for training and testing of the model. All pre-processing steps were performed on data set that was transformed and semi-cleansed. It was observed that all the addresses were combination of all or some of the 5 states (i.e. HOUSE, STREET, SECTOR, ROAD and CITY). 10%, 20% and 40% addresses were selected randomly from dataset for training purpose, and rest of addresses were dedicated for testing through HMM, CRF and their hybrid. Accuracy in terms of corrected segmentation from HMM appeared to be 84.42%, 83.55% and 86.75%; accuracy for CRF accuracy was 87.10%, 89.13% and 88.24%; accuracy for Hybrid-A (HMM followed by CRF) accuracy was 87.15%, 86.34%, 88.40%; accuracy for Hybrid-B (CRF followed by HMM) was 94.50%,96.35%,95.45% given the training percentages as stated above respectively. This shows that increase in training percentage does not make any immense difference in accuracy but it will increases the training effort.

After finding facts from the dataset above, same pre-processing activities were performed with 5000 addresses. All the models (HMM, CRF, Hybrid-A and Hybrid-B) were trained with 10% of data while adhering to tenfold cross validation which gave the average accuracy of 84.52% and 88.60%, 89.22%, 95.50% respectively. Performance of model Hybrid-A and that of model Hybrid-B are comparable but when comes the point of learning cost for converging to more accurate solution, Hybrid-B appears to be a better choice.

It is also observed that increase in size of training would not improve the results. HMM, CRF and Hybrids are fast learning models but they are based on probability so after attaining certain level of accuracy further increasing the size of training set would not affect degree of accuracy. Moreover, concise number of states in a dataset gives more accurate result compared to dataset having more states, though both the datasets have same number of records.

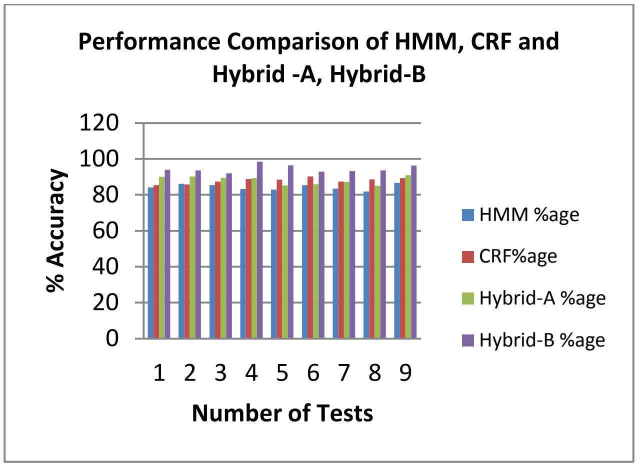

FIG. 9: PERFORMANCE COMPARISON OF CRF, HMM, HYBRID-A AND HYBRID-B TESTS

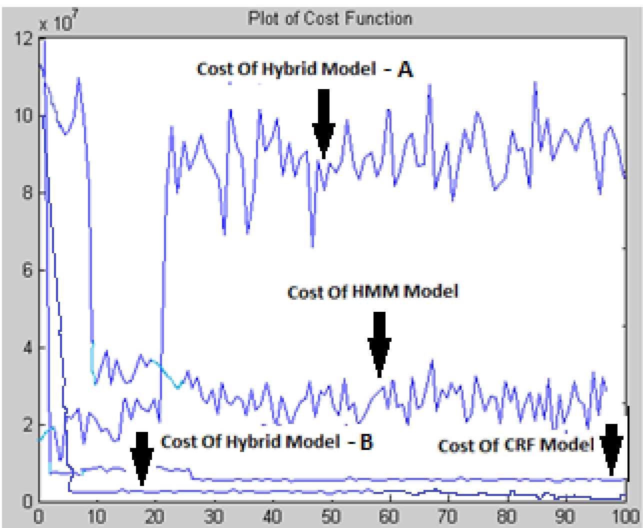

FIG. 10: COST FUNCTION OF CRF, HMM, HYBRID –A AND HYBRID-B TEST

5. Conclusion and Future Work

The proposed model of data cleansing is capable of cleansing any type of large set of data especially addresses. A semi-automatic hybrid mechanism is developed that makes use of two probabilistic machine learning techniques i.e. Conditional Random Fields (CRF) and Hidden Markov Model (HMM). Proposed technique is capable of generating high accuracy with minimum human effort where hybrid of CRF and HMM is found to be more accurate than peer techniques. Another major advantage that can be exploited by using proposed approach is its effectiveness for Asian-style addresses as well as for European-style addresses.

The testing of proposed model can be further improved in various ways such as correction of reference data (addresses) for comparison is made manually and for large dataset, more time will be consumed. There is a need of automatic solution which trains the machines and lessens the training effort. This may be achieved by training of machines by using distinct observation: (based on non-repetitive tags) and / or by setting the weight of the tags.

REFERENCES

2. Vicenç Torra, “Information Fusion - Methods and Aggregation Operators”, 2010, Part 6, pp. 999-1008 11
3. Vinayak R. Borkar, Kaustubh Deshmukh, Sunita Sarawagi. “Automatically Extracting Structure from Free Text Addresses”. Bulletin of the Technical Committee on Data Engineering, 2000, Vol. 23 No. 4 29
4. Mong Li Lee, Hongjun Lu, Tok Wang Ling and Yee Teng Ko. “Cleansing Data for Mining and Warehousing”. Lecture Notes in Computer Science, 1999. 33
5. Hj Mohamed, Tee Leong Kheng, Collin, C., Ong Siong Lee , “E-Clean: A Data Cleaning Framework for Patient Data”, Proceeding of the Informatics and Computational Intelligence (ICI), 2011. 14
6. Vinayak R. Borkar, Kaustubh Deshmukh, Sunita Sarawagi.” Automatic segmentation of text into structured records,” Proceedings of the ACM SIGMOD international conference on Management of data, 2001. 24
7. Er. Muheet Ahmed Butt, Er. Majid Zaman, “Data Quality Tools for Data Warehousing: Enterprise Case Study”, IOSR Journal of Engineering (IOSRJEN), 2013, Vol. 3, pp. 75-76. 1
8. www.ctg.albany.edu, Data Quality Tools for Data warehousing- A small Sample Survey, center for technology in government, 2003. 21
9. “Data Cleansing Tools”, www.premier-international.com/solutions_name_and_address_data_ cleansing.aspx, Accessed on July 20, 2013. 27,28
10. “Address cleansing”, www.sqlpower.ca/page/dqguru , Accessed on Aug 5, 2013. 40, 41
11. “PTCL”, www.ptcl.com.pk/pd_content.php?pd_id=41, Accessed on Sep, 2012. 43
12. Erhard Rahm, Hong Hai Do. “Data Cleaning: Problems and Current Approaches” Bulletin of the Technical Committee on Data Engineering, December 2000 Vol. 23 No. 4 30
13. Peter Christen, Daniel Belacic, “Automated probabilistic address standardization and verification”. Australasian Data Mining Conference’, AusDM’05, 2005. 20
14. Mong L. Lee, Tok W. Ling, Hongjun Lu, Yee Ko. “Cleansing Data for Mining and Warehousing”. Proceedings of the 10th International Conference on Database and Expert Systems Applications 1999, pp. 751-760 34
15. Lars Kolb, Andreas Thor, Erhard Rahm “Parallel Sorted Neighborhood Blocking with MapReduce”, Department of Computer Science, University of Leipzig, Germany, 2010. 15
16. Peter Christen ,”Automatic Record Linkage using Seeded Nearest Neighbor and Support Vector Machine Classification”, Proceedings of the ACM SIGKDD conference, Las Vegas, 2008 16
17. Cochinwala, M.; Dalal, S.; Elmagarmid, Ahmed K.; and Verykios, V. S., "Record Matching: Past, Present and Future", Computer Science Technical Reports. Paper 1510, 2001. 23
18. Sander Canisius, Antal van den Bosch, Walter Daelemans, “Discrete versus Probabilistic Sequence

Classifiers for Domain-specific Entity Chunking", 2006. 19
19. Peter Christen, Daniel Belacic, "Automated probabilistic address standardization and verification". Australasian Data Mining Conference', AusDM'05, 2005. 20
20. Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone, "Classification and Regression Trees", Wadsworth, 1984. 39
21. Galhardas Helena, Daniela Florescu, Dennis Shasha, Eric Simon and Cristian-Augustin Saita, "Declarative Data Cleaning: Language, Model, and Algorithms", Proceedings of the 27th International Conference on Very Large Databases, 2001, pp. 371-380. 25
22. Hanna Wallach. "Efficient Training of Conditional Random Fields", In Proceedings of the 6th Annual Computational Linguistics U.K. Research Colloquium (CLUK 6), Edinburgh, U.K., 2003. 22
23. Yongmei Tan. "Applying Conditional Random Fields to Chinese Shallow Parsing", Lecture Notes in Computer Science, 2005 47