ISSN 2278-3091
Volume 9 No.2, March - April 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse204922020.pdf
https://doi.org/10.30534/ijatcse/2020/204922020

A Machine Learning Approach for Cleaning CRM Data

Anurag Deshmukh[1], **Jitendra Singh Thakur**[2]
[1] Department of Computer Science and Engineering, Jabalpur Engineering College, Jabalpur, India,
deshmukh.anurag236@gmail.com
[2] Department of Computer Science and Engineering, Jabalpur Engineering College, Jabalpur, India,
jsthakur@jecjabalpur.ac.in

ABSTRACT

Customer Relation Management (CRM) is important for data savvy companies because their reciprocal growth is determined by the CRM data and other data in periphery of it. A lot of work has been done in the field of CRM like CRM data mining to reveal useful information and customer-centric CRM, but little work has been done for cleaning the CRM data. Many companies struggle with CRM data accuracy at large extent because it changes frequently. This constant change in modern business data requires CRM to be frequently updated in order to stay valid. The frequent updates lead to ambiguity in CRM data. In this paper, the authors present their machine learning approach for CRM data Cleaning that they have implemented as a tool in python. They have validated the output of tool on a real CRM dataset. The tool achieved F-score of 0.96 with the random forest classifier.

Key words : Customer Relation Management, Data Cleaning, Support Vector Machine (SVM), Decision Tree Classifier, Random Forest Classifier.

1. INTRODUCTION

Customer relationship management (CRM) is a method used by companies to manage its valuable customers. Using the CRM data, a company make policies and take important steps to improve its business relationship with customers [14]. Customer Relation Management data plays a vital role in transforming industries [23], as many major decisions taken by industries totally depends on it. It helps companies to satisfy their customers. This leads companies to retain their customers for long time. For many years, companies are accumulating data about customers like emails, phone numbers, addresses etc. These details are prone to change, and the ground level operators in the companies, most of the time, end up making a new entry rather than updating older ones. This all scenario causes ambiguity and redundancy in the customer relation database, and impacts to the serving quality of the company. The end result is that the company loses its customers. A lot of work have been done in field of CRM like many techniques for CRM data mining [18] were introduced, various software with different approaches for customer relation management [15][16][17] are available, but few works are done in cleaning CRM data. There are approaches available for data cleaning [21][22] but none of them provide suitable solution for CRM. The main reason for few works in this field might be the lack of its need in early years but in recent years the number of customers have increased as well as the number of serving companies and this lead a company to take important steps to retain its customer. In this paper the authors present their machine learning approach for cleaning CRM data. In this approach they initially prepared a dataset which process the data to make it suitable for classification and using trained classifier they divided the data into clean and redundant datasets. Further application of rules on the redundant dataset cleans it and then the clean entries are merged in the clean dataset. In this manner clean dataset is obtained.

2. LITERATURE REVIEW

The authors searched a lot of resources and could not get sufficient relevant work. There are many proprietary software available online that claims high accuracy and wide variety of option in cleaning CRM data [15][16][17]. But the major problem with the whole process is the hectic approach. Users have to go through a lot of option in order to get a clean file. Some of the option have technical meaning associated with them that make this software harder to use. Talking about the open source software, there are many software for CRM management but almost no option for CRM data Cleaning. Even there is no standard CRM dataset available to work upon. There are few works reported in this area to the best of authors knowledge.

3. TECHNOLOGY USED

3.1. Support Vector Machine

Statistical learning theory and structural risk minimization principle are the basis of Support Vector Machine [1]. It distinguishes the classes with a decision boundary that has the maximum separation distance between the classes [2]. This decision boundary is called the hyperplane. Support Vectors are the points located around the hyperplane. These are the points that play the major role in training [3]. There are several kernels used with the model in support vector Machine. The performance of SVM depends on choice of kernel [4].

Kernel functions

Kernels are used to take data as input and transform it into required form. SVM support following kernels: Linear, Polynomial, RBF (gaussian kernel) and sigmoid.[5][6][7]. Kernel function gives the inner product between the points in a suitable feature space.

Polynomial

The polynomial kernel is a nonstationary kernel and a popular method for nonlinear modelling [5] and can then be written as follows:

$$k(x_i, x_j) = (\gamma x_i^T, x_j + r)^d, \gamma > 0 \qquad (1)$$

where γ is the gamma term in the kernel function for all kernel types except linear, d is the polynomial degree, and r is the bias term in the kernel function.

3.2. Decision Tree Classifier

The Decision tree is the technique of classification. It is used to partition dataset into X classes. Decision tree has two types of nodes, one of them is 'Decision node' and another is 'Leaf node'. Decision node plays major role in generating the test whereas Leaf node represent the output classes [8].

It partitions a data space. Every branch in the tree represents a decision cube. Final results are on the leaf nodes. This method of classification use divide and conquer strategy to continuously partition a data and when making decision it uses greedy approach in order to maximize information gain. While training, it resides the whole data into the memory and the according to the formed structure it makes future decisions [8].

3.3. Random Forest Classifier

Random forest classifier generates different decision trees selected from subset of training set. It generates results by taking in consideration result of each decision tree from the subset. Finally, it generates a vector of all output and chooses the most weighted class as the output of classification [10]. The random classifier selected for this experiment was trained with a training data prepared by selecting some N randomly chosen data points from the dataset [9], and then was used for classification. The classifier automatically generated the forest and generated the result by taking the most popular class into consideration [10]. Information Gain criteria [12] and Gini Index [11] criteria were used as the choice of attribute for measuring the quality. Gini Index in random forest classifier is used as an attribute selection measure. It measures the impurity of an Index. Gini index for a randomly selected entry from the training set stating that it belongs to a class Ci can be written as:

$$\sum\sum_{j\neq i} (f(C_i, T)/|T|)(f(C_j, T)/|T|) \qquad (2)$$

where f (Ci, T)/|T| is the probability that the selected case belongs to class Ci.

Each time a tree is grown to the maximum depth on new training data using a combination of features. Pruning is not done in case of fully-grown trees. This gives random forest lead over other decision tree methods like the method proposed by Quinlan in his work [12].

Performance of the tree-based classifier are mostly based on the pruning methods and they don't have any impact of attribute selection procedure [13]. User intervention is only in selection of feature set for each node and choice of the number of trees to be generated. Then to classify each entry has to pass through the depth each tree involved the randomly generated tree subset of the random forest. At the end the entry is classified into the most weighted class category.

4. PROPOSED APPROACH FOR CRM DATA CLEANING

4.1. Implementation details.

A CRM dataset from a company was requested with terms of not disclosing the dataset in order to test the performance of the approach on a real dataset. All the results mentioned in the paper are obtained on the same dataset.

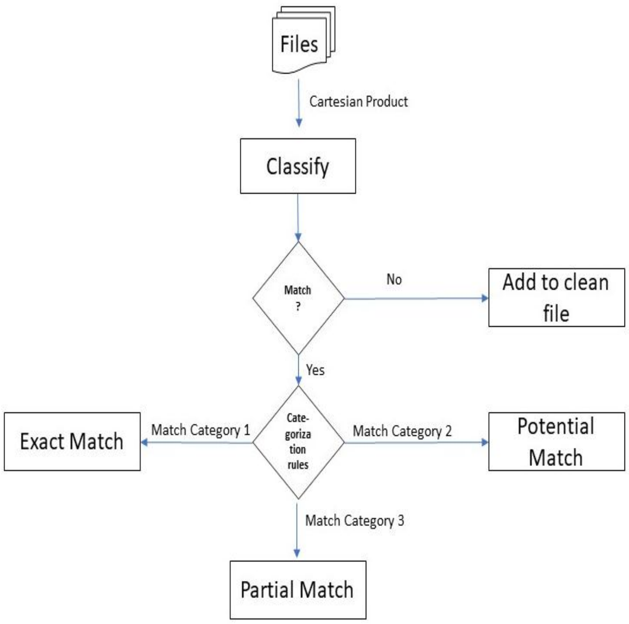

Figure 1 : Flow diagram of approach.

For early training, the whole dataset was pre-processed by filling the missing value and encoding the data into binarized values. During this whole process, distribution of dataset was kept in consideration. Various training cases and labels were prepared before binarization of dataset and few dirty data of different type like missing name, missing email, mismatching email, format change of phone number etc. was introduced in the dataset. One more feature named Y was added into the dataset denoting the label of the column as match or no match in binarized manner. The obtained dataset from the step was later divided into 3 disjoint datasets. Training dataset, Test dataset and Validation data set in ratio of 60:20:20. While distribution, random entries were selected to form each of the dataset. Training dataset was used to train various classifier like SVM with Polynomial kernel, Decision tree, Random forest. Later on, the accuracies of each classifier were tested on test dataset and adjusted accordingly. Final results were tested on the validation dataset.

In between this procedure, various feature set were selected manually and accuracies were tested accordingly. The pattern of overfitting [23] was observed when the model was given large dataset. It was performing well on training dataset but poorly on the test dataset and opposite underfitting [24] pattern was the case when the features set was too small. After all the experiments feature set including name, email, phone number and title was selected as the final feature (feature set 2) set as this set was helping the model to predict matches with higher accuracy on test as well as validation dataset.
After feature set and model selection the whole dataset was divided into two categories of clean data and match data using the trained classifier. The matched dataset was analysed with the proposed rules as described in the algorithm section and on the basis of this rules it was divided into 3 categories: Partial match, Potential Match and Exact match. The complete procedure is described in the flow diagram in Figure 1. The Exact match data was the redundant data and it was merged with the clean file by eliminating all the entries expect one from each group observed.

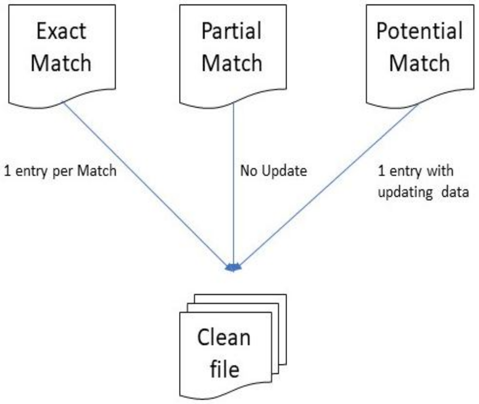

Figure 2 : Merging of obtained files to clean file

In case, of Potential match human intervention is required as the data is displayed by the algorithm and operator decides if it is redundant or not and final entry is the one which is created using all the values present in all the entries. Partial match was as it is merged to the clean file to avoid any data loss. Figure 2 describe the procedure of merging of files. In this manner a clean file can be obtained by supplying a file with redundancies in it.

4.2. Algorithm.

```
1. READ: Contact.csv as df
2. Select feature_set. Say feature set 2
3. COMPUTE: df -> df_encoded
     3.1.  COMPUTE: l = len(df)
     3.2.  Take cartesian product of the file(l*l).
       COMPUTE: df -> df_per
           n12, n21 = percentage name match (1 with 2 and 2
           with 1)
           ep=email match percentage
           pp=phone number matching
           t12, t21= title matching percentage
     3.3.  Encode(df_per):
           FOR i in feature_set:
             FOR val IN df_per[i]
                   IF val >=  90:
                      RETURN 4
                   ELSE IF val >=  75:
                      RETURN 3
                   ELSE IF val >=  50:
                      RETURN 2:
                   ELSE IF val >=  25:
                      RETURN 1
                   ELSE:
                      RETURN 0
                   ENDIF
             ENDFOR
           ENDFOR

4. COMPUTE: classifier(df_encoded)
       OUTPUT:
           redundant_file
           clean_file
5. COMPUTE: Categorization(redundant_file)
       READ: redundant_file
       IF (n12 == 4 or n12 == None) AND (n21 == 4 or n21 ==
       None) and (ep == 4 or ep == None) AND (pp == 4 or pp
       == None) and (t12 == 4 or t12 == None) AND ( t21 ==
       4 or t21 == None):
           PRINT: "Exact Match"
           Add one entry to the Exact_match_file

       ELSE IF (n12 > 2 and n21 > 2) and (ep == 4 or pp == 4
       or (a12 == 4 and a21 == 4)):
           PRINT:" Potential Match"
           Merge all entries and create updated entry
           Add updated entry to Potential_match_file
```

```
    ELSE:
        PRINT: “Partial Match”

        Add to both entries to Partial_match file
    ENDIF

6. COMPUTE: Clean_file
    READ: Exact_match_file, Potential_match_file,
    Partial_match_file, Clean_file
    MERGE: (Exact_match_file, Potential_match_file,
    Partial_match_file, Clean_file) => Clean_file

END
```

5. EVALUATION OF APPROACH

As described in the implementation section various features with classifiers were tried in order to obtain best results. Below are the results of some of the experiments. The dataset used in the evaluation for this approach was CRM of a company, this dataset was asked for evaluating the algorithm with a term of non-disclosure of the data. Below are the results of trained classifier predicting match and no match and their accuracy on the validation data set prepared. Here feature set 1 consist of Address, Name, Phone Number, Email, Title whereas feature set 2 consist of Name, Phone number, Email, Title.

5.1. Results of different classifier with featuere set 1 and feature set 2

Random forest with feature set 1

The combination of random forest model along with the fields Address, Name, Email, Phone No, Title in feature set gives result mentioned in Table 1.

Table 1 : Results obtained by combination of Random Forest classifier and feature set including Address, Name, E-mail, Phone Number, Title.

S.no	Total entries	Redundant	True Positive	True Negative	False Positive	False Negative
1	49	10	4	39	0	6
2	200	25	12	175	0	13
3	1000	100	36	900	0	64
4	2000	100	36	1900	0	64

SVM with Polynomial Kernel with feature set 2

All the kernels available for SVM were tried like Linear, sigmoid, polynomial. But there was no remarkable result. One of them is the combination of SVM with Polynomial kernel along with the fields Name, Email, Phone No, Title in feature set gives result mentioned in Table 2.

Table 2 : Results obtained by combination of SVM classifier with polynomial kernel and feature set including Name, E-mail, Phone Number, Title.

S.no	Total entries	Redundant	True Positive	True Negative	False Positive	False Negative
1	49	10	1	0	39	9
2	200	25	1	0	175	24
3	1000	100	1	0	900	99
4	2000	100	1	0	1900	99

Decision Tree Classifier with feature set 2

With Decision tree classifier as model and the same feature set the result was quite satisfactory but still it was dropping some of very close matches in large dataset. Table 3 denotes the results with Decision Tree classifier.

Table 3 : Results obtained by combination of Decision Tree classifier and feature set including Name, E-mail, Phone Number, Title

S.no	Total entries	Redundant	True Positive	True Negative	False Positive	False Negative
1	49	10	10	39	0	0
2	200	25	20	175	0	5
3	1000	100	92	900	0	8
4	2000	100	90	1900	0	10

Random Forest Classifier with feature set 2

The result obtained by using Random forest as the model and Name, Email, Phone No, Title as the feature set were close to the best possible result of the experiments. Table 4 contains the result obtained from the combination.

Table 4 : Results obtained by combination of Random Forest classifier and feature set including Name, E-mail, Phone Number, Title.

S.no	Total entries	Redundant	True Positive	True Negative	False Positive	False Negative
1	49	10	10	39	0	0
2	200	25	24	175	0	1
3	1000	100	96	900	0	4
4	2000	100	94	1900	0	6

Model vs Accuracy graph

The accuracy on the graph is the average of the prediction of the combination overs all the dataset it is tested on. Figure 3 is the graph obtained as the result of the calculation.

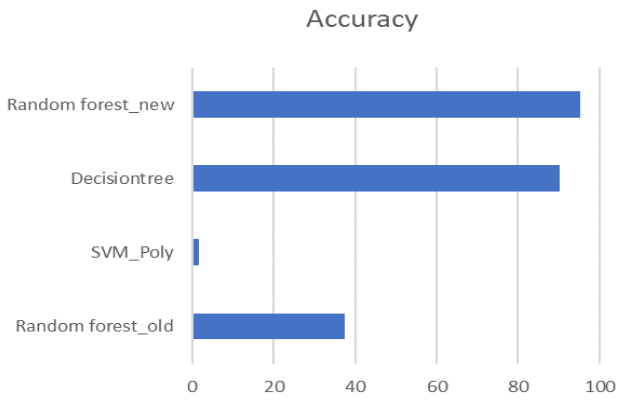

Figure 3 : Accuracy graph of different classifiers.

Table 5 contains the accuracy values obtained after analysis of different models.

Table 5 : Accuracy of each classifier on the dataset.

Model	Accuracy
Random Forest with feature set 1	37.44681
SVM_Poly with feature set 2	1.702128
Decision Tree with feature set 2	90.21277
Random Forest with feature set 3	95.31915

feature set 1 consist of Address, Name, Phone Number, Email, Title and feature set 2 consist of Name, Phone number, Email, Title.

Model vs F-score graph

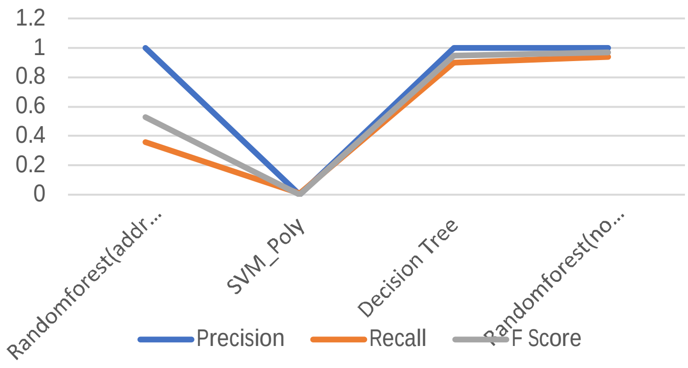

Figure 4 : F-score comparison of all classifiers.

Figure 4 is the plot of precision, recall and f-score of all models used and Table 6 contains the values obtained in analysis of all models.

Table 6 : Precision, Recall and F score of all the models. Here feature set 1 consist of Address, Name, Phone Number, Email, Title whereas feature set 2 consist of Name, Phone number, Email, Title.

S.no	Model	Precision	Recall	F score
1.	Random Forest Classifier with feature set 1	1	0.36	0.529412
2.	SVM with Polynomial kernel with feature set 2	0.000526	0.01	0.001
3.	Decision Tree Classifier with feature set 2	1	0.9	0.947368
4.	Random Forest Classifier with feature set 2	1	0.94	0.969072

6. CONCLUSION

In this paper, the authors presented a machine learning approach for CRM data cleaning. The authors experimented with three machine learning models viz. SVM, Decision Tree and Random Forest. The results showed that the random forest model along with the proposed categorization method performed well in cleaning CRM data with F-score of 0.96. The results also suggest that choosing proper number of features in the feature set is also very important as the accuracy changes with the change in the number of features in the training dataset.

7. FUTURE WORK

As open source CRM datasets are not available at present, the following enhancements can be done in the proposed work if some large open source CRM datasets are available in future:

1. Feature set selection is an important aspect of the algorithm. With the availability of larger CRM datasets, this part can be automatized by developing algorithms that automatically selects the best set of features, and later forms later rules according to the feature set.
2. Partial matches generated in the process are left without any processing due lack of abundant data. These entries are the most difficult to separate and need sufficient data to find a suitable solution.
3. Human intervention is required in case of potential match to select the required entry. With availability of large dataset this intervention can be minimized to large extent by training an algorithm for creating a most updated entry.
4. Since the algorithm is little time consuming, multithreading would resolve the issue. The complete algorithm can be redesigned to work faster by dividing files in chunks and processing them parallelly.

REFERENCES

1. Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik. "A training algorithm for optimal margin classifiers." *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*. 2003.
2. Abe, Shigeo. "Two-class support vector machines." *Support Vector Machines for Pattern Classification*. Springer, London, 2010. 21-112. https://doi.org/10.1007/978-1-84996-098-4_2
3. Muñoz-Marí, Jordi, et al. "Semisupervised one-class support vector machines for classification of remote sensing data." *IEEE transactions on geoscience and remote sensing* 48.8 (2010): 3188-3197. https://doi.org/10.1109/TGRS.2010.2045764
4. Xu, Chong, et al. "GIS-based support vector machine modeling of earthquake-triggered landslide susceptibility in the Jianjiang River watershed, China." *Geomorphology* 145 (2012): 70-80. https://doi.org/10.1016/j.geomorph.2011.12.040
5. Gunn, Steve R. "Support vector machines for classification and regression." *ISIS technical report* 14.1 (1998): 5-16.
6. wei Hsu, Chih, Chihchung Chang, and Chihjen Lin. "A practical guide to support vector classification." *National Taiwan University, Taiwan, Tech. Rep* (2010).
7. Pradhan B (2012) A comparative study on the predictive ability of the decision tree, support vector machine and neuro-fuzzy models in landslide susceptibility mapping using GIS.computers& Geosciences 51(2013):350–365. https://doi.org/10.1016/j.cageo.2012.08.023
8. Liu, Bing, Yiyuan Xia, and Philip S. Yu. "Clustering through decision tree construction." *Proceedings of the ninth international conference on Information and knowledge management*. ACM, 2000.
9. Breiman, Leo. "Bagging predictors." *Machine learning* 24.2 (1996): 123-140. https://doi.org/10.1007/BF00058655
10. Breiman, Leo. "Random forests." *UC Berkeley TR567* (1999).
11. Breiman, Leo. *Classification and regression trees*. Routledge, 2017. https://doi.org/10.1201/9781315139470
12. Quinlan, J. Ross. *C4. 5: programs for machine learning*. Elsevier, 2014.
13. Mingers, John. "An empirical comparison of pruning methods for decisiontree induction." *Machine learning* 4.2 (1989): 227-243. https://doi.org/10.1023/A:1022604100933
14. Wikipedia contributors. "Customer relationship management." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 4 Dec. 2019. Web. 10 Dec. 2019.
15. Winpure, Data-deduplication, https://winpure.com/ deduplication-software.html, Accessed 9 Nov 2019.
16. Dedupe.Io,http://dedupe.io, Accessed on 9 Nov 2019.
17. Strategicdb, https://strategicdb.com/data-cleansing-services/deduping-tool,Accessed 9 Nov 2019.
18. Stefanou, Constantinos J., Christos Sarmaniotis, and Amalia Stafyla. "CRM and customer-centric knowledge management: an empirical research." *Business Process Management Journal* 9.5 (2003): 617-634. https://doi.org/10.4324/9780080472430
19. Buttle, Francis. *Customer relationship management*. Routledge, 2004.
20. Rahm, Erhard, and Hong Hai Do. "Data cleaning: Problems and current approaches." *IEEE Data Eng. Bull.* 23.4 (2000): 3-13.
21. Zhang, Shichao, Chengqi Zhang, and Qiang Yang. "Data preparation for data mining." *Applied artificial intelligence* 17.5-6 (2003): 375-381.
22. Yaghoubi, Maryam, Hamed Asgari, and Marzieh Javadi. "The impact of the customer relationship management on organizational productivity, customer trust and satisfaction by using the structural equation model: A study in the Iranian hospitals." *Journal of education and health promotion* 6 (2017). https://doi.org/10.4103/jehp.jehp_32_14
23. Hawkins, Douglas M. "The problem of overfitting." *Journal of chemical information and computer sciences* 44.1 (2004): 1-12. https://doi.org/10.1021/ci0342472
24. Molinara, Mario, Maria Teresa Ricamato, and Francesco Tortorella. "Facing imbalanced classes through aggregation of classifiers." *14th international conference on image analysis and processing (ICIAP 2007)*. IEEE, 2007. https://doi.org/10.1109/ICIAP.2007.4362755