Using Unlabeled Data Set for Mining Knowledge from DDB

Azhar F. Hassan
Department of Computer Science, Al Nahrain University, Baghdad, Iraq

ARTICLE INFO

Article history:

Received February 28, 2021
Revised March 15, 2021
Accepted March 24, 2021

Keywords:

Knowledge Discovery in Databases (KDD);
Data Mining (DM);
Clustering Data Mining;
Classification Data Mining;
Distributed Database (DDB);
Distributed Data Mining (DDM)

ABSTRACT

In this paper, two algorithms were introduced to describe two algorithms to describe and compare the applying of the proposed technique in the two types of the distributed database system. The First Proposed Algorithm is Homogeneous Distributed Clustering for Classification (HOMDC4C), which aims to learn a classification model from unlabeled datasets distributed homogenously over the network. This is done by building a local clustering model on the datasets distributed over three sites in the network and then build a local classification model based on labeled data products from the clustering model. In the one computer considered as a control computer, we build a global classification model and then use this model in the future predictive. The Second Proposed Algorithm in Heterogeneous Distributed Clustering for Classification (HETDC4C) aims to build a classification model over unlabeled datasets distributed heterogeneously over sites of the network, the datasets in this algorithm collected in one central computer and then build the clustering model and then classification model. The objective of this work is to use the unlabeled data to introduce a set of labeled data that are useful for build a classification model that can predict any unlabeled instance based on that classification model. This was done by using the Clustering for Classification technique. Then presented this technique in distributed database environment to reduce the execution time and storage space that is required.

Azhar F. Hassan
Department of Computer Science, Al Nahrain University, Baghdad, Iraq.
Email: azhar.flaih@ced.nahrainuniv.edu.iq

1. INTRODUCTION

Data mining and Knowledge Discovery in Databases (KDD) have become commercially important techniques and active areas of research in recent years [1, 2]. Because of the growth in database size and usability. Knowledge Discovery in Databases, often called data mining has emerged in the 1990s as a visible research area with the explicit goal of developing tools to facilitate the discovery of higher-level information or knowledge in large databases [3-5]. Data mining is the extraction of non-trivial knowledge from databases using algorithms from computer science and other disciplines. There are several Data Mining tasks, including classification, regression, clustering, association rules, etc. each of these tasks can be regarded as a kind of problem to be solved by Data Mining. The first step in designing a data mining algorithm is to define which tasks the algorithm will address [6-8]. Business applications of data mining software are commonplace and are commodities in many cases. However, data mining of technical data is still a relatively disorganized discipline compared to business applications of data mining. Current data mining procedures have been successful with business applications such as market basket analysis. However, as data mining of technical data becomes important in such technical areas as medicine and engineering, the potential costs of errors will require the data miner to consider other algorithms in addition to the commonly used algorithms [9, 10]. The information discovered is often expressed as a family of classification rules that allow one to classify unseen objects, that is, to predict the behavior of objects other than the ones stored on the database. Through this process of analysis and prediction, we can optimize decision-making tasks [11]. In this work, we try to use unlabeled datasets by presented various existing techniques from machine learning and econometrics for learning from unlabeled data.

2. RESEARCH METHOD

Many situations present huge volumes of raw data, but assigning classes is expensive because it requires human insight. We drew a sharp distinction between supervised and unsupervised learning, classification, and clustering. But what if the goal is classification, but input contains unlabeled data. You can't classify without labeled data. It would be enormously attractive to be able to leverage a large pool of unlabeled data to obtain excellent performance from it. This technique is called the clustering for classification technique. Bayesian techniques offer very useful information about the mutual dependencies among the features in the application domain. Using probabilities rather than hard decisions seems beneficial because it allows the procedure to converge slowly instead of jumping to conclusions that may be wrong. Such information can be used for gaining a better understanding of the dynamics of the process under observation. It seems a good example to apply it as classification algorithms. In addition to using simple partitioned clustering methods called K-Means clustering techniques, here we proposed a new technique to do this using dataset that is geographically distributed as one process of distributed data mining (DDM) processes.

2.1. Homogeneous Distributed Clustering for Classification algorithm (HOMDC4C)

DDM must deal with different possibilities of data distribution; different sites may contain data for a common set of features of the problem domain. In the case of relational data, this would mean a consistent database schema across all the sites. This is a homogeneous case. We present an algorithm to produce a classifier from clustering the data at each site in separate local clustering models and then combine these models; this manner is the cheapest and quickest but often the least accurate solution. This algorithm is called Homogeneous Distributed Clustering for Classification algorithm (HOMDC4C) [12, 13]. It generally consists of three main steps: (a) local clustering, (b) combination, (c) global classification. Here we explain each step in more details:

A. Local clustering: in each site in the network, we will build a local clustering model based on the local data set. By applying the K-Means clustering algorithm in each site separately depend on a fixed number of clusters and the same initial selected centers.

B. Combination of the probability: combine the local computation of the prior probability of each class and the posterior probability of samples conditioned on classes from each site to the central site and collect these values to average it in the central site.

C. Global classification: the last step of our algorithm is to build a naïve Bayesian classification model on the set of labeled data results in the central site and find the posterior probability to each class in the data set. In this case, we can find the class label to any instant of data which class label is unknown by finding the maximum posterior probability to this instance. The (HOMDC4C) proposed algorithm in all details.

```
Algorithm 1: (HOMDC4C) proposed algorithm

Input: DB contains data sample distributed over the network;
Number of clusters L;

Output: classifier model that predicts the cluster of any sample X;

Method:
(1)  Arbitrary choose same L objects as the initial cluster centers;
(2)  Repeat
(3)  (Re) assign each object to the cluster to which the object is the most similar based on the mean
     value of the object in the cluster at each site;
(4)  Compute the new cluster means;
(5)  Until no changes;
(6)  Given an unknown data sample X;
(7)  Compute the prior and conditional probability to X in each site
(8)  Collect this probability in a central site and find these averages in the central site
(9)  Find the posterior probability P (Ci) and conditional P(X|Ci) to X
(10) Find the maximum P (Ci|X) for L classes, C1, C2… Cl to sample X, called the posterior
     hypothesis, as in (1).
```

$$P(Ci|X) = P(X|Ci)\, P(Ci)/P(X) \quad (1)$$

By applying (HOMDC4C) algorithm, the data in these statues will be distributed over the sites in the network, the central site in the network will be sent to the initial centers that choose by the manager in the site1 to other sites which contain the sets of data.

Step (A). Local Clustering: Each site converts this data from unlabeled data to labeled data. This process is done by the clustering technique. Which obtain by assigning each object to the cluster with the nearest centroid. In this application, we measure the distance between each value in the sum field with the two centroids selected

Step (B). Computation of the probability: we need to compute the posterior probability for the train datasets, P (Ci|X), as in the equation in the (HOMDC4C) algorithm by collecting the probability from each site. This by computing the P(X|Ci) the posterior probability of X conditioned on Ci and the prior probability P(Ci), which is independent of X, based on the data in each site in the network.

Step (C). Global classification: by using the values in each of the columns of site1 (central site), we obtain the following posterior probabilities for each possible classification for the unseen instance. Let see the first one and classify it:

Sunny Hot High Weak

Class=yes
P (X| play tennis=”yes”) =0.15*0.2*0.36*0.63=0.007
P (X| play tennis=”yes”) P (play tennis=”yes”) =0.007*0.56 =0.004

Class=no
P (X| play tennis=”no”) =0.6*0.46*0.84*0.31=0.072
P (X| play tennis=”no”) P (play tennis=”no”) =0.072*0.43 =0.04

Therefore, the HOMDC4C algorithm classifier predicts play tennis on this day = “no,” this result is true compared with the human decision. In this way, we can find all other test datasets in Table 1.

Table 1. Test dataset with the class predictive and actual class

Scenario	Outlook	Temperature	Humidity	Wind	Predict class
Day1	Sunny	Hot	High	Weak	No
Day2	Sunny	Hot	High	Strong	No
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	Yes
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	No
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes

2.2. Heterogeneous Distributed Clustering for Classification algorithm (HetDC4C):

Mining distributed data constitutes an important class of (DDM) problems. In the general case, the data sites may be heterogeneous. In other words, sites may contain tables with different schemata. Different features are observed at different sites. We would like to mention that a heterogeneous database, in general, could be more complicated than the above scenario. The approach to building a model from distributed heterogeneous data set is to aggregate all unlabeled data to the central site and build a clustering model on the merged data sets. This model is based on the idea of the K-Means clustering algorithm to obtain labeled data that is useful in building a more accurate classifier model that presented as a naïve Bayesian classifier. This method of DDM is simple and efficient when the amount of geographically distributed data is very small. Although having all the data available typically gives the most accurate predictive models, this approach may be too expensive for several reasons. First, it increases the network traffic dramatically and, in addition, may result in problems related to data delivery, cleaning, and aggregation. Secondly, the data processing algorithm complexity may become an issue [12, 14]. This algorithm is called Heterogeneous Distributed Clustering for Classification algorithm (HETDC4C). It consists of three main steps: (a) collection of the data, (b) clustering model, (c) classification model. Here we explain each step in more details:

A. Collection of the data: collect all the data from the entire site in the distributed database to a single large repository in the central site to begin the process.
B. Clustering model: we will build a clustering model on all the collected unlabeled data by applying the L-Means clustering algorithm to provide a set of labeled data for the classification step.
C. Classification model: Finally, we build a naïve Bayesian classification model on the set of labeled data results from the clustering model and find the posterior probability to each class in the data set. In this case, we can find the class label to any instant of data which class label is unknown by finding the maximum posterior probability to this instance.

Algorithm 2 illustrates the (HETDCFC) proposed algorithm in all details.

Algorithm 2 (HETDC4C) proposed algorithm

```
Input: DB contains data sample distributed over the network;
          Number of clusters N;
Output: classifier model that predicts the cluster of any sample X;
Method:

(1)  send the set of data that are conforming the central schema to the central site at each site;
(2)  arbitrary choose N objects as the initial cluster centers at the central site;
(3)  repeat
(4)  (re)assign each object to the cluster to which the object is the most similar based on the mean value
     of the object in each cluster;
(5)  compute the new cluster means;
(6)  until no change
(7)  given an unknown data sample X;
(8)  find the posterior probability P(Ci) and conditional P(X|Ci) to X
(9)  Find the maximum P (Ci|X) for n classes, C1, C2… Cn to sample X, called the posterior
     hypothesis, as in (2).
```

$$P\ (Ci|X) = P\ (X|Ci)\ P\ (Ci)/P(X) \tag{2}$$

By applying the HETDC4C algorithm, In the case of a heterogeneous distributed database, we must do some preparation to the train datasets before applying the algorithm on it. This process is illustrating in the following steps:

Step (A). Collection of the data: collect all the data from the three sites (site2, site3, site4) to the central site (site1) in the distributed database as the single large repository—a condition by the schema of a table in the central site.
Step (B). Clustering model: we will build a clustering model on all the collected unlabeled data applying the L-Means clustering algorithm. To provide a set of labeled data for the classification step, doing this under the same number of clusters and initial centroids as appeared (centroide1=91, centroid 2=123).
Step (C). Classification model: we build a naïve Bayesian classification model on the training dataset in the central site, which finds.

Then it can see that this value more accurate than the value in (site1 central site) because the data is collected in one site and then extract the posterior probability from it. From this table, we can find the class label to any instant of data which class label is unknown by finding the maximum posterior probability to this instance. Take the test dataset and find the predictive class for each instance depending on the value. Obtain the same value in the predictive class field.

2.3 Performance for predictive model

We can measure the performance of the resulted model by the primary source of performance measurements coincidence matrix. The equations that can be calculated from the coincidence matrix are also given as

True positive rate =TP/TP+FN
True Negative Rate =TN/TN+FP
Accuracy = TP+TN/TP+TN+FP+FN

True positives (TP): The prediction was yes, and the true value is yes
True negatives (TN): The prediction was no, and the true value is no
False positives (FP): The prediction was yes, but the true value was no
False negatives (FN): The prediction was no, but the true value is yes
The overall accuracy of a classifier is estimated by dividing the total correctly classified positives and negatives by the total number of samples.
We can find compute the overall accuracy of the classifier to the above algorithm
True positive rate =7/7+0 = 1.00
True negative rate = 3/3+0 =1.00
Accuracy =7+3/7+3+0+0 = 100%

2.4 Implementation Environment

We conducted an analytical study as well as experimental evaluations on real datasets for the proposed algorithms in terms of correctness, efficiency. In this experiment, we compared HOMDC4C and HETDC4C algorithm as the accuracy of the algorithm and time spent in executing it. The database that used in the proposed system is of weather cases collected from several places of weather forecasting in Iraq of 30,000 records distributed at three sites, 10,000 records for each site, then the results compared. And we use about 100 records as labeled data to estimate the accuracy of the classifiers and compare the results of three produce classifiers. This dataset is a weather case study, and the target is “can play tennis,” this dataset consists of 7 attributes (fields of data) and two classes (yes or no). The proposed algorithm is evaluated using a coincidence matrix. These experiments were implemented in a distributed environment by using a small LAN network that included four computers, each computer having a constant IP address. Three of which are applied as three different branches of the company at different locations and the last is applied as a control site. Each computer has at least the probability of Pentium 4 with 2.1-2.7 GHz, 512 MB RAM, and Visual FoxPro 9.0 that was used to build the program and distributed it over the network.

3. RESULTS AND DISCUSSION

The availability of vast amounts of data by applications has made imperative the need to use unlabeled data. This is because the cost of assigning labels to all the data can be expensive, and/or some of the data might not have any labels. Incorporating unlabeled data might not always be useful, so in recent years various techniques have been proposed for utilizing unlabeled data. The underlying challenge is to formulate a learning task that uses unlabeled data by presented various existing techniques from machine learning and econometrics for learning from unlabeled data. In this paper, we try to use the unlabeled data to learn the supervised classification techniques by introduced the Clustering for Classification technique that implementing a clustering model on the unlabeled data to introduce a set of labeled data that are useful for build a classification model that can predict any unlabeled instance based on that classification model. We presented this technique in distributed database environment to reduce the execution time and storage space that is required; this technique is implemented in two scenarios of distributed as homogeneous and heterogeneous and compared the performance of each technique against the performance of a traditional supervised classifier in measures of accuracy result, timely execution and communication overhead. We tried to include datasets from various domains with different characteristics to make the study as generalizable as possible.

3.1. Experiments

In this section, we compare the execution of two proposed algorithms in a real-world environment which consisted of a distributed database system that consists of three branches separate at different sites, three sites. There are 10,000 records in each site and 100 records required to estimate the accurately in a central site. The results are detailed as follows.

1. (HOMDCFC) algorithm:
   The time and the accuracy that can be obtained by applying this algorithm is:
   Step1: Build local clustering models in each site in the network,
   Step2: Build a classification model in each site.
      1. Estimate the accuracy of the global classifier that is (0.90%) by applying the Confusion Matrix
      2. Three forms are required to execute this algorithm. Then it's more complex than (HETDCFC).

2. (HTEDCFC) algorithm:
   Step1: Collecting data sets in time as in Table 2.
   Step2: Build a clustering model in time about (2.457) per second.

Step3: Build a classification model in time about (3.468) per second.

1. Estimate the accuracy of the classifier that is about (0.95%).
2. One form is required to execute this algorithm which is simpler than (HOMDC4C) algorithm.

Table 2. Times of executing the collecting of data from each site

Time per second spent in each site	
No of records	**Time of collecting**
1000	1.500
2000	1.688
3000	2.375
4000	3.812
5000	4.800
6000	5.346
7000	6.844
8000	6.456
9000	6.203
10000	9.969

3.2. Comparison between HOMDC4C and HETDC4C

The difference between the two proposed algorithms (HOMDC4C and HETDC4C) to show the efficiency obtained from the two algorithms in criteria of the accuracy of the classifier, time of execution algorithm, cost require for central storage. These criteria are shown in the general case in Table 3. By the process of measuring the time console in the execution of the algorithms, Fig. 1 shows the chart of the execution time measure in second with the number of data entry to the algorithm range initially from having 1000 records of datasets in every three sites in-network and the increase by 1000 records in each step accessible to 10000 records in each site. i.e., 30000 records enter the two algorithms. Fig. 2 shows the accuracy of the two proposed algorithms in percentage per the number of records that are used to measure these accuracy ranging from 10 records increased by ten up to 100 records.

Table 3. Comparison between two algorithms

DDM Tech.	Time of execution	Accuracy	Cost of communication
HOMDCFC	Low	Low	Low
HETDCFC	More	More	More

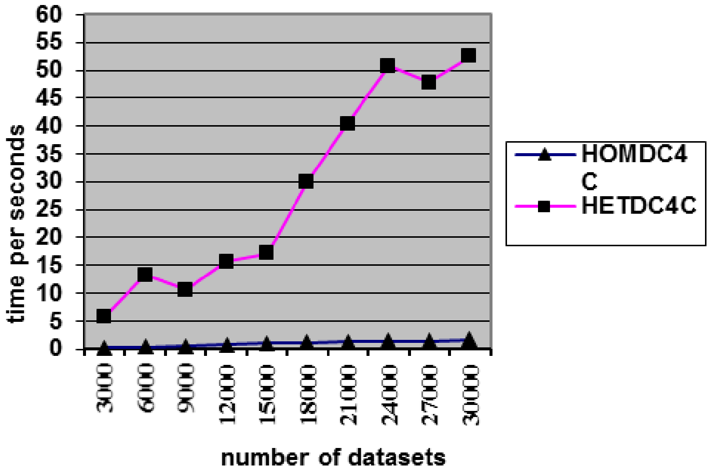

Fig. 1. The execution time of the two proposed algorithms in seconds

4. CONCLUSION

The conclusions are drawn from implementing the proposed algorithm in the real world and comparing its results with those that are obtained from the most famous traditional classification technique (Naive Bayesian) algorithm. In this paper, we analyzed a newly proposed technique (C4C) to solve the problem of using unlabeled data in a supervised classification algorithm, this technique based on applying any clustering algorithm on unlabeled data to produce a set of labeled data which is used to build the classifier model that can predicate any unseen instance sample. Using a Naïve Bayesian algorithm as a classification algorithm rather than another algorithm seems beneficial because it uses probabilities instead of hard decisions that allow the procedure to converge slowly to solve instead of jumping to conclusions that may be wrong. We estimated that the (HOMDC4C) proposed algorithm more a cheap and quickest but often the least accurate solution from the compeer (HETDC4C) proposed algorithm. We used the environment of the distributed database in this algorithm to reduce the time that spends in communicating and execution time in the proposed technique.

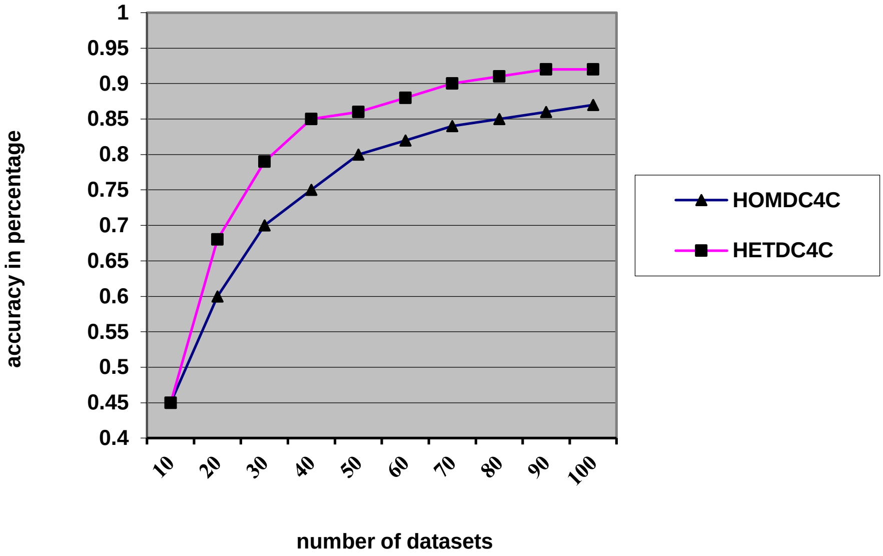

Fig. 2. The accuracy of the two proposed algorithms per record number

REFERENCES

[1] O. Maimon and L. Rokach, "Introduction to knowledge discovery and data mining," in *Data mining and knowledge discovery handbook*: Springer, 2009, pp. 1-15. https://doi.org/10.1007/978-0-387-09823-4_1

[2] A. Dogan and D. J. E. S. w. A. Birant, "Machine learning and data mining in manufacturing," *Expert Systems with Applications*, vol. 166, pp. 114060, 2021. https://doi.org/10.1016/j.eswa.2020.114060

[3] A. Y. Sun and B. R. J. E. R. L. Scanlon, "How can Big Data and machine learning benefit environment and water management: a survey of methods, applications, and future directions," vol. 14, no. 7, pp. 073001, 2019. https://doi.org/10.1088/1748-9326/ab1b7d

[4] P. Burggräf, J. Wagner, and T. J. E. S. w. A. X. Weißer, "Knowledge-based problem solving in physical product development—A methodological review," vol. 5, pp. 100025, 2020. https://doi.org/10.1016/j.eswax.2020.100025

[5] O. H. Yahya, H. T. S. Alrikabi, and I. A. Aljazaery, "Reducing the data rate in internet of things applications by using wireless sensor network," *International Journal of online and biomedical engineering,* Article vol. 16, no. 3, pp. 107-116, 2020. https://doi.org/10.3991/ijoe.v16i03.13021

[6] S. Darrab, D. Broneske, G. J. I. J. o. M. L. Saake, and Computing, "Modern Applications and Challenges for Rare Itemset Mining," vol. 11, no. 3, 2021.

[7] M. Al-dabag, H. S. ALRikabi, and R. Al-Nima, "Anticipating Atrial Fibrillation Signal Using Efficient Algorithm," *International Journal of Online and Biomedical Engineering (IJOE),* vol. 17, no. 2, pp. 106-120, 2021. https://doi.org/10.3991/ijoe.v17i02.19183

[8] N. S. Alseelawi, E. K. Adnan, H. T. Hazim, H. Alrikabi, and K. Nasser, "Design and Implementation of an E-learning Platform Using N-Tier Architecture," *International Journal of Interactive Mobile Technologies,* vol. 14, no. 6, pp. 171-185, 2020.

[9] R. Nisbet, J. Elder, and G. Miner, *Handbook of statistical analysis and data mining applications*. Academic Press, 2009.

[10] N. A. H. Hala A. Naman, Mohand Lokman Al-dabag, Haider Th. Salim Alrikabi, "Encryption System for Hiding Information Based on Internet of Things," *International Journal of Interactive Mobile Technologies (iJIM),* vol. 15, no. 2, 2021. https://doi.org/10.3991/ijim.v15i02.19869

[11] T. Kliegr, Š. Bahník, and J. J. A. B. S. Fürnkranz, "Advances in machine learning for the behavioral sciences," vol. 64, no. 2, pp. 145-175, 2020. https://doi.org/10.1177/0002764219859639

[12] A. S. Rostami, M. Badkoobe, F. Mohanna, A. A. R. Hosseinabadi, and A. K. J. T. J. o. S. Sangaiah, "Survey on clustering in heterogeneous and homogeneous wireless sensor networks," vol. 74, no. 1, pp. 277-323, 2018. https://doi.org/10.1007/s11227-017-2128-1

[13] W. Gan, J. C. W. Lin, H. C. Chao, J. J. W. I. R. D. M. Zhan, and K. Discovery, "Data mining in distributed environment: a survey," vol. 7, no. 6, pp. e1216, 2017. https://doi.org/10.1002/widm.1216

[14] G. Pio, F. Serafino, D. Malerba, and M. J. I. s. Ceci, "Multi-type clustering and classification from heterogeneous networks," vol. 425, pp. 107-126, 2018. https://doi.org/10.1016/j.ins.2017.10.021

BIOGRAPHY OF AUTHORS

Azhar F. Hassan
She finished her undergraduate program at the college of Baghdad in Computer Science. Her master's program was from the University of Technology in Computer Science.