Volume 8, No. 5, May-June 2017
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Probabilistic Threshold Query on Uncertain Data using SVM

Kavita Shevale
Research Scholar, BharatiVidyapeeth Deemed University
College of Engineering, Pune, India

Prof. Gajanan Bhole
Assitatnt Professor, BharatiVidyapeeth Deemed University
College of Engineering, Pune, India

Prof. Milind Gayakwad
Bharati vidyapeeth Deemed University
College of Engineering, pune India

Abstract: Data needed for fired query is available on the internet only concern is how effectively and efficiently it is delivered to the end user. This task is not easy because size of data is rapidly increasing and cost that we spend on correct data should be lesser than the value of data to be searched. There is one more very important factor to be considered is uncertainty of data. Uncertainty of data is nothing but percentage of correctness in the result. Uncertainty of data may cause a hurdle in searching correct or desired data. To answer this type of query probabilistic approach is useful where the extent of accuracy is calculated. This accuracy calculation using probabilistic approach is very important to decide usefulness of the data. This may differentiate between very useful data a hardly useful data.Paper concentrate on the probabilistic approach where the Support Vector Machine is utilized for the classification of a data; experimentally it has been proved that approach utilized delivers superior results than the approach where Enhanced Learning Machine is used.

Keywords: Probability Threshold query, Support Vector Machine, Enhanced Learning Machine

1. INTRODUCTION

User can only enter the data for which he or she is looking for, there is no direct interface or mechanism in existing search engine to control the percentage of the results. Also, it is not expected from users to know about percentage of accuracy or results while searching. Most of the time search engine sorts the result according to the descending order of the percentage of accuracy and top 10-12 results are displayed to the user. This is possible in search engine because of number of algorithms and supporting softwares but if we compare the same scenario especially with relational database, we have least freedom while firing the query to deal with the percentage and comparatively less support at the database management system's end to deal with it.

Moreover, there is one more fundamental concern in the process known as uncertainty in the database. Uncertainty [1][2]in a database is because of noise introduced in the database this could be simply considered as a noise in the database. This is of two types tuple level uncertainty of a data and row level uncertainty of data. In case of tuple level uncertainty only concern is whether to use the respective tuple or not. In case of the attribute level uncertainty of data is comparatively sever because we never know the number columns to be included rather which column is to be considered.

Table 1: Selection of Job Aspirant

Sr. No.	Percentage	Score in the test	Score in Certification exam of DBMS
1	70%	80%	90%
2	80%	75%	85%
3	75%	90%	79%

Consider a scenario where the company organizes recruitment drive for selection of employee from a campus of college. To accomplish this selection process company considers some important criterion like Percentage acquired by the students in the curriculum, Score gained by students in the test held by the organization and Certification exam on a specific database management system. [7][8][9]

Now the problem is none of the student tops all three forms of scores neither student scores low in all three forms of the score to filter out the weak performer. This type of situation can be considered with the attribute level uncertainty where there is no rule or clear way to select certain result with exact specification of the percentage. In this case that organization may prefer any one of the criteria with top priority and the selection could be accordingly or there could be combination of more than one attribute and processing on that result like calculation of an average could be done. In this situation, it is not clear attribute or attributes influencing the selection.

Consider a same scenario of recruitment process where there are some selections to be made from cities Mumbai, pune and delhi; interview process is over but based on the current requirements few selections are to be put on hold. There is requirement of resources at pune so the question remains whether to delete students selected from Delhi and Mumbai. This is an example of tuple level uncertainty, here the uncertainty is about inclusion of tuples.

Probabilistic database is dealing with the accuracy of the data so the extension probability threshold query deals with the accuracy of the data and a mechanism where we can consider the results purely based on the accuracy but not on the basis of just query. For example, if an organization is looking for students to recruited with percentage greater than 60 and city should be pune, what about candidates with 70% or more and ready to relocate at pune. This is the result of probability threshold query where threshold is manipulated to deliver more superior results.

Paper covers various approaches [6] to classify data and algorithms to process on the data especially ELM and SVM.

2. LITERATURE SURVEY

Several research papers referred with aim to improvise performance of query in terms of time by keeping eye on accuracy. Probability Threshold Query, is influenced by the type of query used that is Skyline Query, Nearest Neighbourhood, Revers Nearest Neighbourhood, Top K queries.

Skyline query is a type of a query where condition or intersection is observed; it is suggested that it is not necessary to maintain the rigidity of threshold it can be adjusted as per the significant occurrence of results[5,3,2].

Nearest neighbour approach gives the results which gives the result in the descending order of nearness. That is if we consider the same example, in that case Nearest neighbour approach will return results with descending value of quality and ascending value of cost.

KNN gives the one more filtration level to the Nearest neighbour approach by allowing to provide, For example, list of 10 best hotels in the world. value k, which is nothing but the positive number. Top K [4] approach works on same phenomenon it delivers all number of tuples, with filtration.

ReverseNearest neighbour [10] is an approach which returns all the values satisfying the result. This means, ReverseNearest neighbour will return all the hotels considering the condition that values must be part of the system.

This analysis will make the task of algorithm working on query, for setting the limit of threshold correctly. Approach to be utilized for processing should be fast to cope up with effective processing of a query dynamically. There are possibly good options like ensemble Algorithm, Adaboost Algorithm, Support Vector Machine Algorithm, Decision Tree algorithm. These approaches might be useful not only for learning but also for classification.

3. SYSTEM ARCHITECTURE

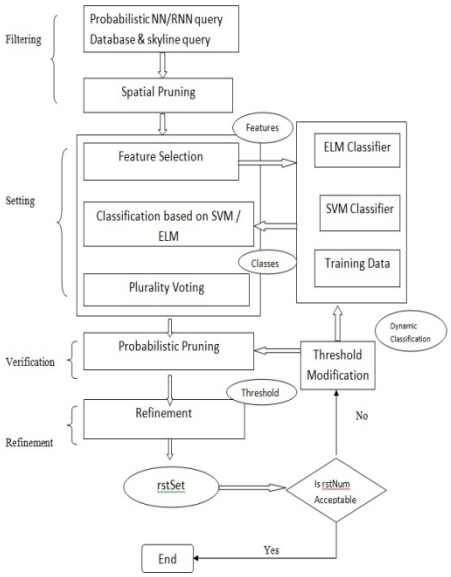

Figure 2: System Architecture

3.1 Modules:

There are four phases present in this architecture.

1. Filtering
2. Setting Phase
3. Verification
4. Refinement

1. Filtering:

In the filtering phase, all the objects which do not have chance to being a result of query are removed here. Generally, for improving the process of filtering, the spatial pruning algorithm uses the spatial locations. The objects which are not pruned insert the cndSet (candidate object set) and then transfer to the next phase.

Filtering process shortlists those tuples with lowest probability of being member of a set of selected tuples. Spatial pruning algorithm helps to improve the performance of filtering process. Tuples which are not pruned are forwarded for next phase.

2. The setting phase:

The setting phase is the set of s-threshold λ for query q which is based on classification threshold, which is the main work of this proposed paper. In this phase, first the feature values of q which are useful are selected and calculated here. After that ELM classifier and SVM Classifier predicts the threshold class of q, it should be corresponding class, where method of plurality voting is applied. At the last of this phase, threshold value of the q predicted in this phase is transfer to the next phase.

3. Verification phase:

The verification phases confirm that which object will satisfies or fails the PTQ's. Here, for calculating the lower or upper probability bounds several probabilistic pruning algorithms might be proposed. Those objects which are not rejected or accepted are stored in the rfnSet and then transfer to the next phase.

4. Refinement Phase:

This is the last phase of the architecture. In this phase, the correct probability of every object in the rfnSet is calculated to whether the final result set or not rstSet. If the number of rstSet is suitable, and the query is ended; or else, the threshold must be modified and recall the phase three. In addition, during the process if dynamic environment is detected which will be based on a sliding window methods, the ELM classifier and SVM classifier must be retrained.

There is need to select the approach to efficiently deliver the result. This is simple harmonic mean of precision and recall. Number 2 denotes balancing factor.

$$\textbf{Efficiency} = \frac{2*\text{precision}*\text{recall}}{\text{precision} + \text{recall}}$$

Where,

$$\textbf{Precision} = \frac{\text{desired result} \wedge \text{fetched result}}{\text{Fetched Results}}$$

Precision is the probability of finding desired results in among fetched results

$$\textbf{Recall} = \frac{\text{desired result} \wedge \text{fetched result}}{\text{Desired Results}}$$

Recall is probability of finding, how many fetched results are desired.

4. RESULTS

Experiment is performed to check the performance of skyline queries and Probabilistic queries using SVM and ELM based on the parameters like Time, Precision and Recall and observation of these experiments are stated below.

Performance of Skyline queries using ELM and SVM:

After performing the experiment on skyline query values of precision, recall is similar but there is significant change in the time taken to build a model is concerned. ELM consumes

0.27 seconds and SVM 0.05 seconds. More experiments are needed to note observations and comment about performance of SVM; so, performance of the SVM and ELM is compared with probabilistic queries.

Skyline Query			
Classifier	Time taken to build model in sec	Precision avg.	Recall in avg.
ELM	0.27	0.525	0.725
SVM	0.05	0.525	0.725

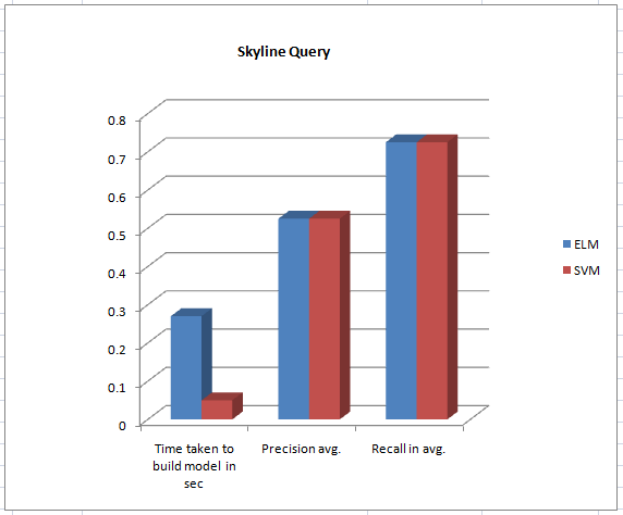

Figure2: Performance of Skyline queries using ELM and SVM

After performing the experiment on probabilistic queries values of precision, recall is similar but there is significant change in the time taken to build a model is concerned. ELM consumes 0.42 seconds and SVM 0.05 seconds.

Probablistic Query			
Classifier	Time taken to build model in Sec	Precision avg.	Recall avg
ELM	0.42	1	1
SVM	0.05	1	1

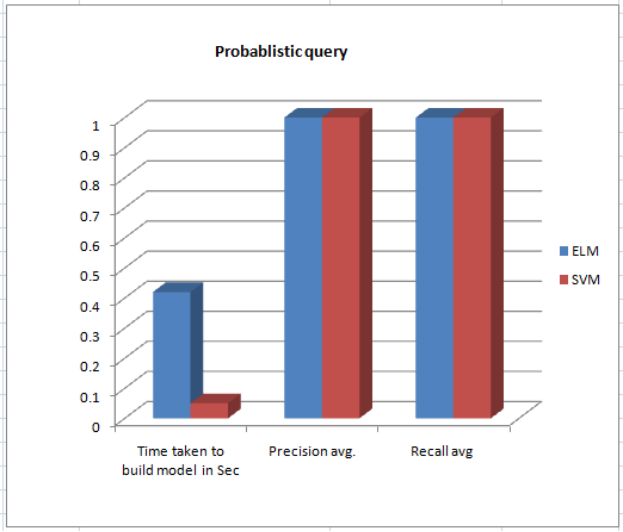

Figure3: Performance of probabilistic queries using ELM and SVM

This clearly shows the efficiency of the SVM algorithm over ELM whether it is skyline query or probabilistic query.

5. CONCLUSION

Experiment performed proves the superiority of Support Vector Machine by delivering the consistent results as far as time complexity is concerned.

As we can see for SVM using probabilistic query saves 0.37 seconds and SVM using skyline query saves 0.22 seconds. When we consider more number of queries more amount of time will be saved.

6. REFERENCES

[1] R.Cheng,J.Chen,M.Mokbel,C.Chow, Probabilistic verifiers: evaluating constrained nearest-neighbor queries over uncertain data, In:ICDE, IEEE, 2008,pp.973–982.

[2] B.Yang,H.Lu,C.S.Jensen, Probabilistic threshold k nearest neighbor queries over moving objects in symbolic in door space, In:EDBT,ACM,2010,pp.335– 346.

[3] X. Lian,L.Chen, Efficient processing of probabilistic reverse nearest neighbor queries over uncertain data,Int.J.VeryLargeDataBases18(3)(2009) 787–808.

[4] M. Cheema, X.Lin,W.Wang,W.Zhang,J.Pei, Probabilistic reverse nearest neighbor queries on uncertain data,Trans.Knowl.DataEng.22(4)(2010) 550–564.

[5] T.Bernecker, T.Emrich,H.Kriegel,M.Renz,S.Zankl,A.Züfle, Efficient probabilistic reverse nearest neighbor query processing on uncertain data, Very LargeDataBases4(10)(2011)669–680.

[6] J.Li,B.Wang,G.Wang,Efficient probabilistic reverse k- nearest neighbors query processing on uncertain data, in: Database Systems for Advanced Applications, Springer,2013,pp.456–471.

[7] G.-B.Huang,Q.-Y.Zhu,C.-K.Siew, Extreme learning machine: a new learning scheme offeed forward neural net works, in:2004IEEE International Joint Conference on Neural Networks,2004.Proceedings,vol.2,IEEE,2004, pp. 985–990.

[8] G.-B. Huang, Q.-Y.Zhu,C.-K.Siew, Extreme learning machine: theory and applications, Neuro computing70(1)(2006)489–501.

[9] G.-B. Huang,D.H.Wang,Y.Lan, Extreme learning machines: asurvey, Int.J. Mach. Learn. Cybern.2(2)(2011)107–122. [17] R. Cheng, D.Kalashnikov,S.Prabhakar, Querying imprecise attain moving object environments, Trans. Knowl.DataEng.16(9)(2004)1112–1127.

[10] R.Cheng,X.Xie,M.L.Yiu,J.Chen,L.Sun,Uv-diagram: avoronoi diagram for uncertain data,in:ICDE,IEEE,2010, pp.796–807.