Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 7, July 2015, pg.212 – 218

RESEARCH ARTICLE

PERFORMANCE ANALYSIS OF HYBRID APPROACH OF K-NN ALGORITHM USING MULTIPLE-LEVEL LEARNING FOR TEXT CLASSIFICATION

Monika[1], Dr. Rajender Singh Chillar[2]

[1]M.Tech student, DCSA, Maharshi Dayanand University, Rohtak, India

[2]Professor, DCSA, Maharshi Dayanand University, Rohtak, India

Email id- [1] Monika_sarohiwal@yahoo.com

Abstract: -Dataset of each and every institute or organization is rapidly increasing. Due to this, it is going very complex for any user to find and access necessary data from whole dataset. To keep this problem in mind, researchers are working to find a best solution to manage dataset. One of all techniques is k-NN algorithm. This algorithm performed well but time to time new challenges make a need to upgrade this technique. So we choose a large dataset on which we apply modified k-NN technique. Multiple Level Learning is helping technique that improves performance of k-NN.

Keyword: Data Mining, Text Classification, k-NN algorithm, Multiple-Level Learning

1. INTRODUCTION

The increasing computerization with in the world around us has meant that the existence of database containing vast quantities of data is now a fact of everyday life. The enormous quantities of data which are now stored in databases create a problem in that it becomes very difficult to make meaningful sense of such a large quantity of data. For human investigators, the process of extracting meaningful information from such a large amount of data becomes the classic problem of information overload.

The field of data mining seeks to address this problem by the use of computer modeling techniques to derive useful knowledge in a concise form from these large databases. Using the ability of computers to sort ,analyze and categorize large volumes of data extremely quickly, data mining methods seek to redress the problem of information overload by allowing fast and reliable methods of data modeling and representation.

A typical database or dataset to which data mining methods can be applied will consist of a number of data elements or examples, which are termed tuples in the field of relational databases. Each elements or examples of data are made up of

a set of attributes, each of which encodes a value relevant to the type of the given attribute. Generally, all data elements making up a dataset will consist of the same attributes, giving each data element a consistent form. This allows data mining methods to look for patterns and commonality between the data elements over the space of the attributes of the.

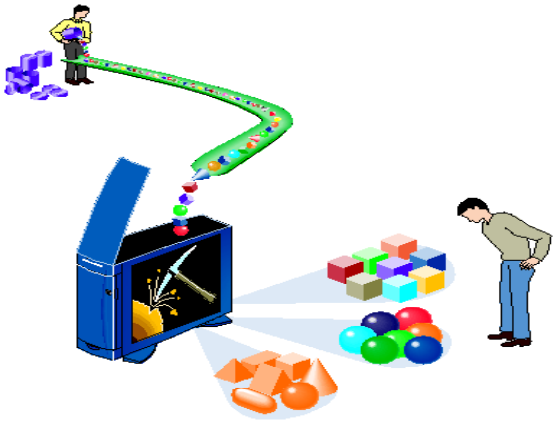


Figure 1: Data Classification

Classification routines in data mining also use a variety of algorithms and the particular algorithm used can affect the way records are classified. A common approach for classifiers is to use decision trees to partition and segment records. New records can be classified by traversing the tree from the root through branches and nodes, to a leaf representing a class. The path a record takes through a decision tree can then be represented as a rule. For example, "Income<$30,000 and age<25, and debt=High, then Default Class=Yes). But due to the sequential nature of the way a decision tree splits records (i.e. the most discriminative attribute-values [e.g. Income] appear early in the tree) can result in a decision tree being overly sensitive to initial splits. Therefore, in evaluating the goodness of fit of a tree, it is important to examine the error rate for each leaf node (proportion of records incorrectly classified). A nice property of decision tree classifiers is that because paths can be expressed as rules, then it becomes possible to use measures for evaluating the usefulness of rules such as Support, Confidence and Lift to also evaluate the usefulness of the tree.

1.1. TEXT CLASSIFICATION

Text classification is one of the major applications of both of these algorithms. Text classification include text classification for Arabic text, Punjabi text, devnagri text, Chinese text, and another a number of other language texts [3]. TC importance rises up since it concerns with natural language text processing and classification using different techniques, in which it makes the retrieval and other text manipulation processes easy to execute.

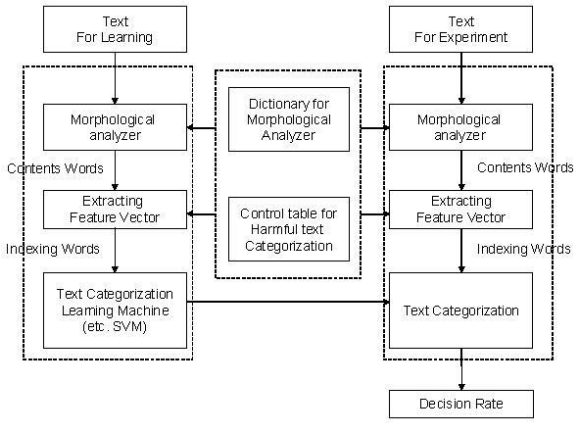


Figure 2 structure of text classification

The objective of text categorization is to assign an entry from a set of predefined categories to a document. So far many methods of the text categorization are presented, such as Support Vector Machine, k nearest neighbors, neural network, bayes classier and decision tree, etc.

1.2. **k-NN CLASSIFICATION ALGORITHM**

The k-nearest-neighbor algorithm is a basic instance based learning method and widely used in similarity classification. The purpose of this algorithm is to classify a new dataset based on attributes and training [4]. The classifier does not use any model to fit and only based on memory. To classify a new pattern x, the k-NN classifiers find k nearest patterns in the training database, and uses the k pattern to determine the class of pattern x.

Given a training set D and a test object x = (x', y'), the algorithm computes the distance (or similarity) between z and the entire training object (x, y) which is belongs to D to determine its nearest- neighbor list, Dz. (x is the data of training object, while y is its class. Likewise, x' is the data of the test object and y' is its class).

Once the nearest-neighbor list is obtained, the test object is classified based on the majority class of its nearest neighbors:

Majority Voting: $y' = \text{argmax} \Sigma (x_i, y_i) I (v = y_i)$

where v is a class label, yi is the class label for the ith nearest neighbors, and I(v=yi) is an indicator function that returns the value '1'if its argument is true and '0' otherwise.

Training algorithm:

Input: - D, the set of k training objects and test object

z= (x', y').

Classification algorithm:

Process: -

1. Compute d (x', x), the distance between z

And every object(x, y) belongs to D.

2. Select Dz (subset of D, the set of k closest

Training objects to z).

Output: - $y' = \arg\max \Sigma (x_i, y_i) I (v=y_i)$

There are several key issues that affect the performance of k-NN [5]. One is the choice of k. If k is too small, then the result can be sensitive to noise points. On the other hand, if k is too large, then the neighborhood may include too many points from other classes.

Another issue is the approach to combining the class labels. The simplest method is to take a majority vote, but this can be a problem if the nearest neighbors vary widely in their distance and the closer neighbors more reliably indicate the class of the object. A more sophisticated approach, which is usually much less sensitive to the choice of k, weights each object's vote by its distance, where the weight factor is often taken to be the reciprocal of the squared distance:

$wi = 1/d (x', xi)2$

This amounts to replacing the last step of the k-NN algorithm with the following:

Distance-Weighted Voting:

$y' = \arg\max \Sigma(x_i,y_i)\ wi*I(v=y_i)$

The choice of the distance measure is another important consideration. Although various measures can be used to compute the distance between two points, the most desirable distance measure is one for which a smaller distance between two objects implies a greater likelihood of having the same class. Thus, for example, if k-NN is being applied to classify documents, then it may be better to use the cosine measure rather than Euclidean distance. Some distance measure can also be affected by the high dimensionality of the data.

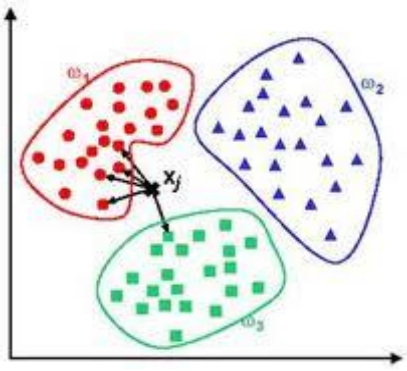


Figure 3 k-NN classifier

2. OBJECTIVE OF RESEARCH WORK

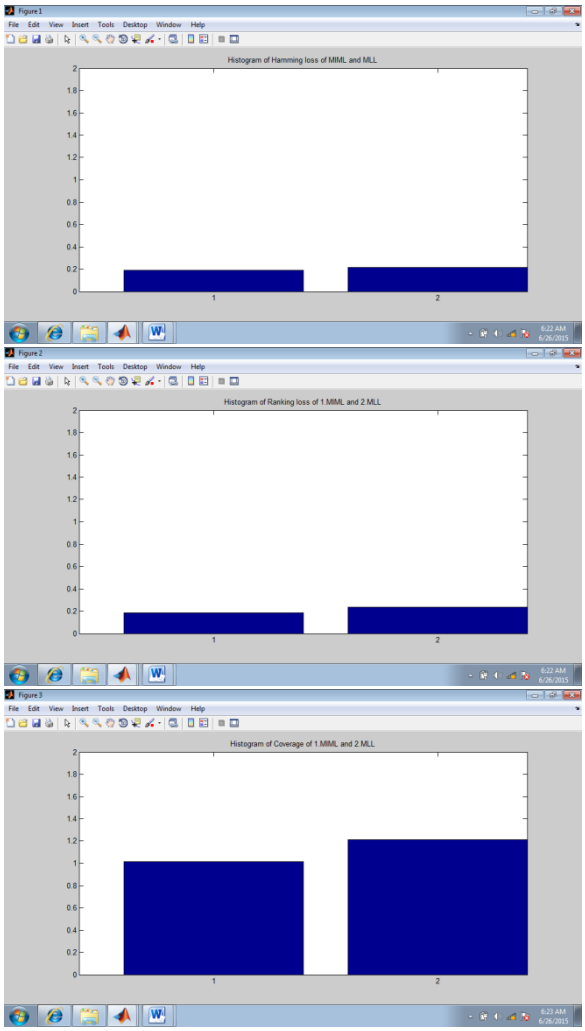
We shall follow these objectives as following:

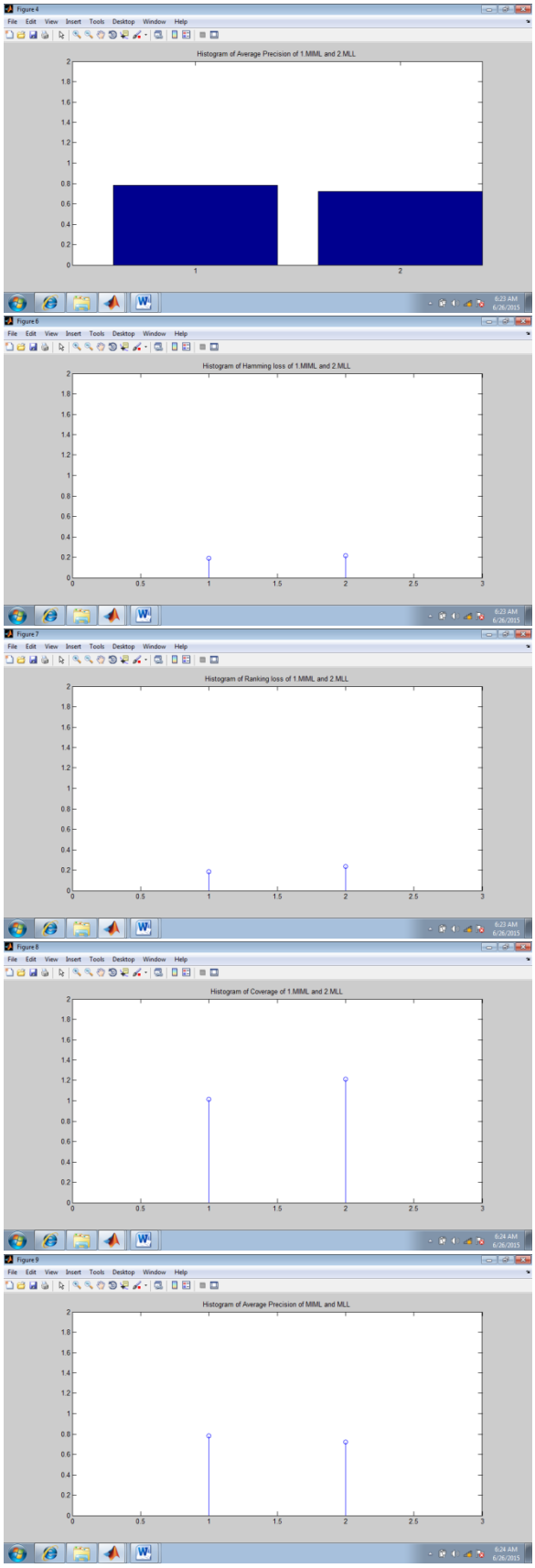
1) Study k-NN algorithm concept in data mining.
2) Calculate parameters value of k-NN like hamming loss, running time, ranking loss, average precision.
3) Use Multiple Label learning with k-NN algorithm.
4) Calculate value of the parameters for this hybrid algorithm.
5) Compare the results.

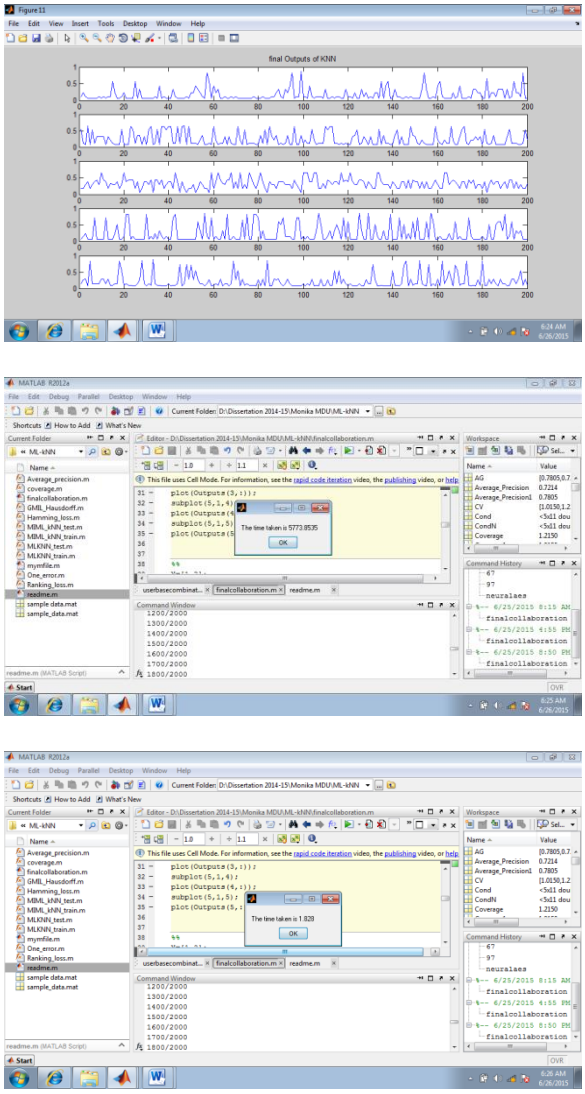
3. PROPOSED METHODOLOGY

Step1: Select data source on which we apply our proposed algorithm.
Step2: Implement k-NN algorithm on selected data source and get performance.
Step3: Now modify k-NN using Multiple Level Learning to develop a hybrid algorithm.
Step4: Hybrid algorithm ready to perform on same data source.
Step5: performance of hybrid algorithm compared with existing k-NN algorithm.
Step6: Our proposal to develop more efficient algorithm that is named hybrid algorithm in this research work.

4. RESULTS







5. CONCLUSION

From all the above calculations we come to the conclusion that the k-NN algorithm is an excellent algorithm when Multiple Level Learning is used with it. K-NNs did an excellent job of ranking individuals for the current dataset. k-NNwas able to do just as well as a complex, well-engineered data mining tool and the handcrafted analysis of a domain expert. It seems likely that in the future more data mining packages will begin to include k-NN with MIML. Although the results of this study appear interesting, it is important to note that they only apply to the current dataset.

ACKNOWLEDGEMENT

Our thanks to IJCSMC for allowing us to modify templates they had developed. Also I take this opportunity to express my special gratitude to my supervisor Dr. Rajender Singh Chillar generously making himself available for her most valuable guidance, encouragement. It would get not be possible for me to make this paper without her guidance.

REFERENCES

[1] J. Han and M. Kamber, "Data Mining Concepts and Techniques", 2nd ed. Amsterdam: Morgan Kaufmann Publishers, 2006.
[2] D. T. LAROSE, "Discovering knowledge in data: an introduction to Data Mining", New Jersey: John Wiley & Sons, 2005.
[3] S. Al-Harbi, A. Almuhareb, A. Al-Thubaity, M. S. Khorsheed, A. AlRajeh," Automatic Arabic text classification" King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia

[4] X. Wu et al. "Top 10 algorithms in data mining" Knowledge information Springer-Verlag London Limited: 22-24, 2007.
[5] Li-Juan Wang, Xiao-Long Wang, Qing-Cai Chen "GA-based Feature Subset Clustering for Combination of Multiple Nearest Neighbor Classifiers" Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005.

