Survey on Sparse Computation for Large-Scale Data Mining

[1]Madhavi Bhamare, [2]Suvidha Patil, [3]Kalpita Kuwar, [4]Sampresha Shinde

[1,2,3,4]Department Of Computer, Late. G.N. Sapkal College of Engineering, Nashik, Maharashtra, India.

[1]*bhamaremadhavi@gmail.com* , [2]*suvidha.patil114@gmail.com*, [3]*kuwarkalpita@gmail.com*,

[4]*sampreshashinde@gmail.com*

Abstract - The machine learning methods depends on inputs in the form of pairwise similarities between objects in the data-set. Many pairwise similarities grows quadratically in size of the data-set which poses a challenge like scalability. Efficiency for similarity-based methods is to sporadic the similarity matrix. However, existing methods, firstly compute the complete similarity matrix after that delete some of the non-zero entries. This needs quadratic time and storage and is thus in-docile for big-scale data sets. The method called sparse computation that generates a sporadically similar matrix which contains only same type of methods without performing first all pairwise comparisons. Some similarities are identified by displaying the data onto a low-dimensional space in which set of objects that divide the same grid vicinity are consider of potential high similarity otherwise set of objects that don't divide a neighborhood are considered to be different type and thus their similarities are not calculated. The projection is performed effectively even for massively large data sets. Apply sparse computation for the K-nearest neighbors algorithm(KNN),for graph-based machine learning techniques of supervised normalized cut and K-supervised normalized cut(SNC and KSNC) and for support vector machines with radial basis function kernels(SVM),on real-world classification problems.

Keywords - Big data, Data mining, Similarity-based machine learning, Sparsification, Supervised normalized cut, K-nearest neighbor algorithm.

I. INTRODUCTION

Several leading machine learning techniques for classification and clustering such as the K-nearest neighbor algorithm, Support Vector Machines (SVMs). SEVERAL leading machine learning techniques for classification and clustering such as the K-nearest neighbor algorithm, variants of supervised normalized cut or support vector machines with Gaussian RBF kernels use as input pairwise similarities. The application of similarity-based algorithms to large-scale data sets is challenging because the number of similarities grows quadratically as a function of the number of objects in the data set. Several sparsification approaches known to date, have been applied to reduce the number of non-zero entries in the similarity matrix with minimal effect on specific matrix properties. These approaches, however, have to generate the full set of pairwise similarities in advance and thus take at least quadratic time. In this paper, we propose a novel methodology called sparse computation that overcomes the computational burden of computing all pairwise comparisons between the data points by generating only the relevant similarities. Hence, not only is the resulting matrix sparse but also the computation itself is linear in the number of resulting non-zero entries. The relevant similarities are identified by projecting the data points onto a low-dimensional space in which the concept of grid neighborhoods is employed to devise groups of objects with potentially high similarity. Once the relevant pairs of objects have been identified, their similarity is computed in the

original space. This differentiates the method from known grid-based clustering algorithms that use the grid neighborhoods to identify the clusters. With our approach, objects can belong to the same grid neighborhood while ending up in different clusters, or conversely, belong to different neighborhoods but still get clustered jointly. The grid dimensionality and grid resolution are the parameters that control the density of the generated similarity matrix. A key aspect of sparse computation is the efficient projection of the data onto a low-dimensional space. Well-known methods such as Principal Component Analysis (PCA) or Multidimensional Scaling (MDS) require excessive running times for large and high-dimensional data sets and are thus not practical for large-scale applications. We suggest generating a low-dimensional space using an algorithm referred to here as approximate-PCA. Approximate-PCA provides leading The proposed sparse computation method is broadly applicable for any algorithm that requires the computation of pairwise similarities. Examples of such algorithms include classification algorithms such as the K-nearest neighbor algorithm, variants of supervised normalized cut, support vector machines with non-linear kernels, and spectral methods, as well as clustering algorithms such as Greedy Agglomerative Clustering algorithms, K-means and other Expectation- Maximization algorithms. For graph-based algorithms, such as SNC used here, an additional advantage is that sparse computation tends to break down the data set into a collection of isolated components in the graph. The machine learning task can then be performed for each of these components separately, which leads to further improvement in the efficiency of such data mining algorithms.

The new methodology is applied here to four different similarity-based machine learning techniques: the K-nearest neighbor algorithm (KNN), support vector machines (SVMs) with radial basis function kernels, and two recently devised graph-based machine learning techniques called supervised normalized cut (SNC) and K-supervised normalized.

II. LITERATURE SURVEY

Sparse computation that generates a sparse similarity matrix which contains only relevant similarities without performing first all pairwise comparisons. Sparse computation that overcomes the computational burden of computing all pairwise comparisons between the data points by generating only the relevant similarities.

The classifcation algorithms presented in the previous section, require as input pairwise similarities between the objects in the data set. The number of pairwise similarities grows quadratically in the size of the data set, which poses a challenge in terms of scalability. This challenge is shared also by a vast spectrum of clustering approaches, including greedy agglomerative clustering and expectation maximization algorithms. A great deal of research work has been conducted on sparsifying dense matrices. Such efforts consider input graphs or matrices that are dense and apply sparsifying algorithms that aim to preserve various matrix properties. Arora et al. describe a simple random-sampling based procedure that generates a sparse matrix whose eigenvectors are close to the eigenvectors of the original matrix. The algorithm considers all non-zero entries of the original matrix and uses the Chernoff-Hoeffding bounds to set some of the entries to zero. The running time of this algorithm is at least proportional to the number of non-zero entries in the input matrix. Spielman and Teng present a graph sparsification algorithm that produces a subgraph of the original, whose Laplacian quadratic form is approximately the same as that of the original graph. Their algorithm has a complexity that is close to being linear in the number of non-zero entries in the original Laplacian. Jhurani recently proposed an algorithm that transforms the original matrix into a sparse matrix with minimal changes to the singular values and the singular vectors corresponding to the near null-space of the original matrix. All these sparsification approaches are based on evaluating all entries of the complete similarity matrix and determining for each entry whether or not to round it to zero. The reading of the entries

of the dense similarity matrix alone requires (n2) running time for a data set of n objects. For this reason, these algorithms are not practical for large-scale data sets.

By contrast, our approach determines in advance which entries of the similarity matrix are relevant and evaluates only those. Another strategy that aims to reduce the computational burden of computing all pairwise similarities is proposed. They suggest to use initially an approximate similarity measure to subdivide the objects into overlapping subsets. The exact similarities are then only computed between objects that belong to the same subset. This strategy reduces the running time significantly when the computation of the exact similarity measure is expensive, e.g., when the number of attributes is large. In their paper, McCallum et al. study the problem of reference matching in the context of bibliographic citations of research papers. The problem consists of grouping citations that reference the same paper. The approximate distance measure is based on the number of words two citations have in common, which can be computed efficiently using an inverted index. The complexity of this approach is, however, still (n2) because the approximate similarity measure must be computed for all pairs of objects.

III. SYSTEM ARCHITECTURE

The relevant similarities are identified by projecting the data points onto a low-dimensional space in which the concept of grid neighborhoods is employed to devise groups of objects with potentially high similarity. Once the relevant pairs of objects have been identified, their similarity is computed in the original space. This differentiates the method from known grid-based clustering algorithms that use the grid neighborhoods to identify the clusters. With our approach, objects can belong to the same grid neighborhood while ending up in different clusters, or conversely, belong to different neighborhoods but still get clustered jointly.

-

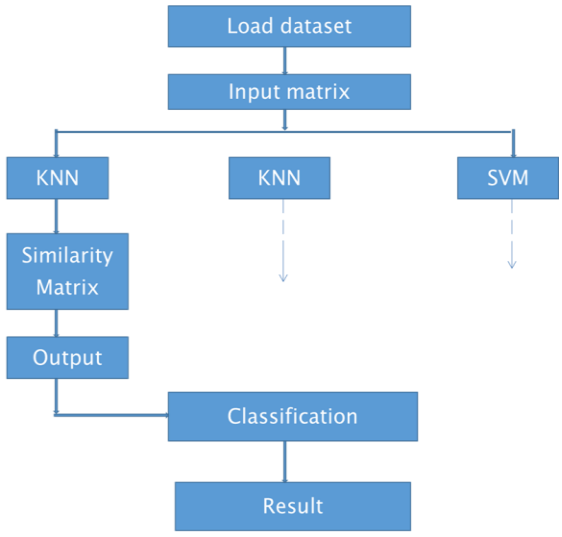

Fig. 1 System Architecture

A key aspect of sparse computation is the efficient projection of the data onto a low-dimensional space. Well-known methods such as Principal Component Analysis (PCA) or Multidimensional Scaling (MDS) require excessive running times for large and high-dimensional data sets and are thus not practical for large-scale applications. We suggest generating a low-dimensional space using an algorithm referred to here as approximate-PCA. Approximate-PCA provides leading The proposed sparse computation method is broadly applicable for any algorithm that requires the computation of pairwise similarities. Examples of such algorithms include classification algorithms such as the K-nearest neighbor algorithm, variants of supervised normalized cut, support vector machines with non-linear kernels, and spectral methods, as well as clustering algorithms such as Greedy Agglomerative Clustering algorithms, K-means and other Expectation- Maximization algorithms. For graph-based algorithms, such as SNC used here, an additional advantage is that sparse computation tends to break down the data set into a collection of isolated components in the graph. The machine learning task can then be performed for each of these components separately, which leads to further improvement in the efficiency of such data mining algorithms.

IV. COMPARISM OF SYSTEM

A. Existing System

The classification algorithms presented in the previous section, require as input pairwise similarities between the objects in the data set. The number of pairwise similarities grows quadratically in the size of the data set, which poses a challenge in terms of scalability. This challenge is shared also by a vast spectrum of clustering approaches, including greedy agglomerative clustering and expectation-maximization algorithms.

A great deal of research work has been conducted on sparsifying dense matrices. Such efforts consider input graphs or matrices that are dense and apply sparsifying algorithms that aim to preserve various matrix properties.

B. Proposed System

To improve the performance of similarity based algorithm for large scale data set in data mining. This system generates only the relevant similarities without performing all pairwise comparisons between objects in the data set using KNearest Neighbor algorithm (KNN).

V. CONCLUSION

Similarity-based algorithms have not been used for large scale data mining. Here, sparse computation, which provides practical efficiency for similarity-based algorithms while retaining their performance. The method generates only the relevant similarities without performing all pairwise comparisons between objects in the data set.

ACKNOWLEDGMENTS

It gives us great pleasure in presenting the Survey Paper on "Survey on Sparse Computation for Large-Scale data mining." I would like to take this opportunity to thank my internal guide Prof. S. B. Wagh for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their Valuable suggestion were very helpful

REFERENCES

[1] Dorit S. Hochbaum, Philipp Baumann, "*Sparse Computation for Large-Scale Data Mining*", IEEE Transactions on Big Data.

[2] T.M. Cover and P.E. Hart, "*Nearest neighbor pattern classification*," IEEE Trans. on Information Theory, vol. 13, pp. 21–27, 1967.

[3] D.S. Hochbaum, C.-N. Hsu, and Y.T. Yang, "*Ranking of multidimensional drug profiling data by fractional-adjusted bi-partitional scores*," Bioinformatics, vol. 28, pp. i106–i114, 2012.

[4] Y.T. Yang, B. Fishbain, D.S. Hochbaum, E.B. Norman, and E. Swanberg, "*The supervised normalized cut method for detecting, classifying, and identifying special nuclear materials*," INFORMS Journal on Computing, 2013.

[5] D.S. Hochbaum, C. Lu, and E. Bertelli, "*Evaluating performance of image segmentation criteria and techniques*," EURO Journal on Computational Optimization, vol. 1, pp. 155–180, 2013.

[6] B. Sch¨olkopf and A.J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Cambridge MA: MIT Press, 2001.

[7] P. Baumann, D.S. Hochbaum, and Y.T. Yang, "*A comparative study of leading machine learning techniques and two new algorithms*," 2015, submitted 2015.

AUTHORS PROFILE

Author Name :- Madhavi Bhamare

Qualification :- Diploma in Computer Engg from Mumbai University, B.E Appear of computer Engineering department from Late G. N. College Of Engineering Savitribai Phule Pune University.

Author Name :- Kalpita Kuwar

Qualification :- Diploma in Computer Engg from Mumbai University, B.E Appear of computer Engineering department from Late G. N. College Of Engineering Savitribai Phule Pune University.

Author Name:- Suvidha Patil

Qualification:- Diploma in Computer Engg from Government Polytechnic Nasik, B.E Appear of computer Engineering department from Late G. N. College Of Engineering Savitribai Phule Pune University.

Author Name : Sampresha Shinde

Qualification :- Diploma in Computer Engg from Mumbai University, B.E Appear of computer Engineering department from Late G. N. College Of Engineering Savitribai Phule Pune University.

