FP- International Journal of Computer Science Research (IJCSR)
Volume 2, Issue 2, Pages 160-166, June 2015

To Enhance A-KNN Algorithm for Improving Software Architecture

Poojia Gupta
Department of Computer Scienceand Engineering
Chandigarh University, Mohali, Punjab, India
pujagupta008@gmail.com

Gurpreet Kaur
Department of Computer Scienceand Engineering
Chandigarh University, Mohali, Punjab, India
gurpreetcse.cgc@gmail.com

ABSTRACT

Software Architecture is important factor for the development of complex and big software system. Software Architecture Decomposition is an important part in software design. Software clustering is used to cluster functions of similar type in one cluster and other are in other cluster. K-mean is the base of the clustering but it has some limitations. A-KNN, WPGMA, UWPGMA, CLINK and SLINK these clustering methods are used for decomposition the software architecture. A-KNN cluster method is more efficient than others methods but some time functions are highly coupled then cluster technique does not find out correct distance. So that need to enhancement in Euclidian distance formula based on normalization. In this paper, an enhancement has been proposed in the Euclidean distance formula which increased the cluster quality. When the cluster quality will be increase we are able to properly cluster the function and improve the software architecture and A-KNN will be generate the best results than previous method. It has covered the concept of cohesion and coupling between components of the system. Reduce the complexity of the system.

General Terms: Clustering Algorithms, Software Architecture

Keywords: K-mean, KNN, A-KNN, Clustering, Decomposition, Software Architecture.

1. INTRODUCTION

System engineering is concern with deployment, architectural design and integration where as software engineering is concern with development, quality, testing and control of the system. Software engineering is an approach to the development, design operation and maintenance of software and its main focus in on to assure the quality of the product, detect and prevent system from bugs by testing or analysis. In software design the main responsibility of software architect to handle an issue or other tasks that make software design more difficult. Software specialists grasp strategy regarding their work using a couple of methods, methodology and gadgets depending on the benefits

accessible and issue to be understood. It is the procedure of tackling client's issues by the deliberate improvement and progression of expensive, great programming frameworks inside of expense, time and different obliges [6]. Software designing is about arrangement of ventures to deliver the product, from its introductory stage to its final stage. A software engineering is related with all the angles that are utilized as a part of the product generation or make the product. Categorization of software into two classes: System Software and the Application Software. The system software is utilized for organizing the equipment segments. The software encompasses the requests and working course of action to handle all equipment components. It provides the data concerning components. It is stage for application to finish the errand. System software controls the general arrangement of programming. The application software used for to achieve the specific assignments. It could or could not encompass the solitary system.

Software Architecture: Software Design mentions to the elevated level constructions of a software system. Software design is one of the most vital accomplishment factors for the progress of convoluted and large software systems. The design is a construction of the arrangement that contain the software agent, external viable properties of that agent and connection amid those components.

1. Architecture is constituent and connector.

2. An architecture is the building block of any system that include units, that possess the behavior of the system, connections show how units transfer information or share information, and connections also show in which direction program operations are activated either sequentially or concurrently [7].

3. Architecture including frank things.

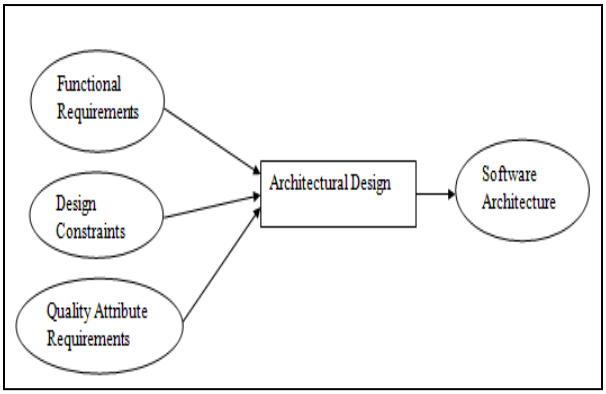


Fig 1: Software Architectural Design [7]

Clustering in software Engineering: Software clustering is a vital train in software engineering and software maintenance It manages the software unsupervised clustering of software curios like functions, classes, or records into high-level state structures like bundles, segments, or sub-system in light of the likeness of the curios. Software clustering is connected, for occasion, to understand the complex software systems to rebuild software architectures to distinguish reusable parts or to recognize lost software antiques [1]. Early in the historical backdrop of software engineering as an exploration field, it was perceived that the disintegration of an expansive programming system into subsystems was crucial for both the improvement and upkeep period of a software venture. As a general rule, be that as it may, software activities regularly neglect to take the standards of software engineering. This has offered ascent to the compositional recuperation that endeavors to naturally modularize, or bunch, a product framework into significant subsystems [6]. Grouping is characterized as dividing the arrangement of information focuses into subsets such that a particular standard is improved. Every information point is appointed to a group and the basis which is to be advanced is the normal squared separation between the information focuses and its comparing bunch focus [8]. The term grouping is fundamentally confounded with the order. Yet, the couple of thoughts of information investigations that recognized it as takes after: 1.In classification, articles are allocating to officially characterized classes though in clustering the comparative items are gathered in one group and unique items are assembled in another bunch. 2. In clustering the bunch investigation is done and it is not the same as the discriminate examination. The principle point of discriminate examination is to enhance the current clustering by strengthening the class divisions in group investigation there is a need to build the class structure first [8]. The procedure of cluster analysis is utilized to develop little gatherings or bunches having same properties from the substantial heterogeneous arrangement of information. The principal utilization of cluster analysis was accounted in London. Amid cholera erupt in London in 1854; John Snow utilized a unique sort of guide to plotting the instances of infections that were accounted for. The last perception that was seen after the formation of the guide was that there was a nearby union between the thickness of ailment cases and a solitary well pump that was situated at a focal point of the road. After that, the well pump was ousted and putting an end to the infection. The relationship between the phenomena is generally harder to recognize, yet this was conceivable with the assistance of cluster analysis. This is the first and basic use of cluster analysis.

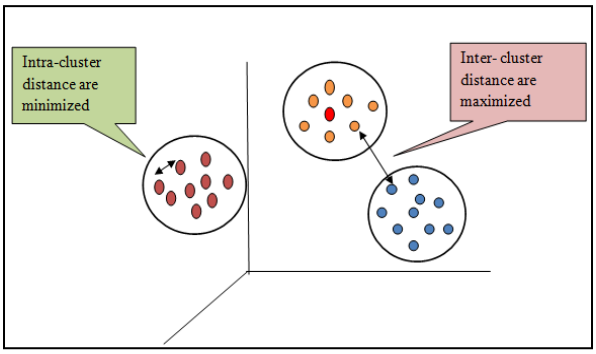


Fig 2: Concept of Software clustering [8]

K-mean Clustering: In 1967, MacQueen early counseled K-mean clustering algorithm. K-mean algorithm is one of the accepted partitioning algorithms. The idea is to categorize the data into k clusters whereas k is the input parameter enumerated in advance through iterative relocation method that converges to the innate minimum. It consists of two distinct periods: Early period is to ascertain k centers at random one for each cluster. The subsequent period is to ascertain distance amid data points in Dataset and the cluster centers and allocating the data point to its nearest cluster. Euclidean distance is usually believed to determine the distance. When all the data points are encompassed in little clusters an early grouping is done. New centers are next computed by seizing the average of points in the clusters. This is done because of inclusion of new points could lead to a change in cluster centers. This procedure of center updation is iterated till a situation whereas centers do not notify anymore or criterion the function becomes minimum. This signifies the convergence criteria. The main aim of k-mean clustering is to minimize the Sum of Square Error (SSE). Mainly the squared error criterion is used:

$$\boldsymbol{E} = \sum_{i=1}^{n} \sum_{x \epsilon Ci} |\boldsymbol{x} - \boldsymbol{mi}| \quad \text{..... Equation 1.1}$$

Where x is the data point and m_i is the center for Cluster C_i. E is the sum of squared error of all the points in the dataset. [9].

Advantages of K-mean algorithm**:**

1. K-mean is robust, fast and easy to understand.
2. It produce best result if data set is large.

Limitations of K-mean algorithm:

1. Early selection of numbers of clusters has to be beforehand known. Because the algorithm does not notify concerning the correctness of the number of clusters, it could be probable to select wrong the value of k.

For example, your data could naturally have five clusters but you fed k= 3, next algorithm will revisit back 3 clusters. Those clusters will not be compact and well separated. If the number of clusters selected is tiny next there could be the chance of allocating dissimilar points in one cluster.

2. It can encompass dead constituent setback that is if cluster center is wrongly selected, it could never be notified and therefore not ever embody a class.

3. Converges to innate minima, i.e. for selected worth of k disparate early centers are chosen randomly by the algorithm on disparate runs [9].

2. RELATED WORK

Mark Shtern and Vassilios Tzerpos [1], In this paper, assorted software clustering algorithms is created alongside its properties, qualities, and confinements. These algorithms utilized for particular specific software system, however, the fundamental component, how to pick a clustering algorithm which is best suitable for detailing of software. In this paper, it gives a method to the picking of software clustering strategy for specific prerequisites. Software clustering is an area that has been creating for quite a while. Countless software clustering algorithms have been proposed. In this paper, concentrate on software clustering methods or systems that gather programming components into subparts. It is essential think to comprehend the software system for somebody. The consequence of a software clustering algorithm is recognized as decomposition of the software. The selection of a software clustering algorithm plays a vital act for crafting the decomposition. This paper gives the strategy to pick the proper algorithm. This methodology permits the assessment of the new algorithm in prior stages before it is executed.

Abdulaziz Alkhalid et al. [2], in this paper they clarified about Software architecture disintegration assumes an imperative part in programming configuration. . Disintegration implies that big or complex issue separated into parts. This paper presents bunching methods for software architecture disintegration. In which two algorithms are used Hierarchical Agglomerative clustering and Adaptive-K nearest neighbor and applied on industrial software system. To calculate the distance between clusters SLINK and WPGMA methods are used. This paper introduces an upgraded methodology for the disintegration of software architecture. It presented the utilization of A-KNN algorithm in software architecture deterioration with the spotlight on functional requirements and their attributes and contrasted its execution with two Agglomerative Clustering algorithm or procedures: SLINK and WPGMA. The outcomes exhibit A-KNN calculation is competitive than two Agglomerative Clustering procedures. It gives priceless data for software designer.

Abdulaziz Alkhalid et al. [3], in this paper an approach is used for finding the solution of two main problems that are face by software designer during software decomposition: 1. it is difficult to determine the number of clusters. 2. For highly coupled component or fuzzy components, it is also difficult to determine the software module or clusters. In this paper fuzzy C-mean clustering (FCM) , three hierarchical clustering technique and Adaptive-K nearest neighbor algorithms are used and these algorithms when applied on real industrial software system conclude that the result produced by A-KNN algorithm is effective and accurate and further FCM gives the valuable information which is used for solving the above two main problems.

Chung-Horng et al. [4], This paper presents a study, in which clustering method was applied on the software system to disintegrate a software architecture based on their functions and their attributes, after disintegration the closely connected requirements are gathered to form a subsystem or module. The disintegration of software system is final step in software designing. A real time network system was used on which various experiments were conducted. In which three similarity coefficients were studied, that is simple matching, Jaccard, Sorensan. When these coefficients were applied on dataset Jaccard and Soresan produced better result .As well as four clustering algorithms are compared WPGMA, UPGMA, SLINK, WLINK, the better result was produced by WPGMA. The result that was produced by WPGMA was very much closer to the result that was predicted by software designer.

Alkhalid et al. [5], this paper introduced the concept of refactoring. The refactoring technique reduces the complexity factor and improved the quality of the software and these are two important issues for software designer. When complexity factor is reduced in refactoring it alter the internal functionality of the system the without affecting its external properties. In this paper, Adaptive K-NN was proposed with its weight attribute. The refactoring technique applied on the function or method level. In this technique three hierarchical agglomerative clustering is used that is

SLINK , CLINK and WPGMA and these techniques are compared with proposed Adaptive K-NN (A-KNN) and all techniques are applied on the industrial software system, the result concluded that the A-KNN clustering technique produced better result as compared to others when refactoring done at the function level.

3. KNN AND A-KNN APPROACH IN SOFTWARE CLUSTERING

The concept of k-NN was first introduced by Fix and Hodges. It is an enhancement of K-mean clustering. It is based upon normalization. KNN is a non parametric lazy learning algorithm. It is very easy to understand but hard to implement. Non-parametric statement means that it does not make any assumptions on the underlying data distribution [3]. It makes decision on the basis of entire training data set. It has minimal training phase but a costly testing phase. Cost is in terms of memory and time. It requires more time to access all the data training sets. It also requires more memory to store all the data. KNN assumes that data is in feature space and data points are metric points. *The* data can be scalars and multidimensional vectors. Since the points are in feature space and has some distance. Each training set is consisting of vector and each vector has label like positive and negative. But KNN works equally with arbitrary numbers. It has value of K. The value of K decides the neighbors of the classification. If the value of K=1 then this algorithm is simply known as nearest neighbor algorithm. It is done to increase the quality of the clustering. It can be used to find out the density estimation of the classification.
KNN approach is basically consisting of following steps:

1. Data set uploads.
2. Find out probabilistic points in which maximum points are lies near centre known as hyper plane.
3. Find out Euclidean distance from hyper plane.
4. Cluster points on the basis of Euclidean distance.

Advantages of K-NN algorithm:

1. It is extremely competent if datasets are large.
2. To training data it is extremely strong as mentioned by their weights.

Disadvantages of K-NN algorithm:

1. First need to ascertain the value of k parameter
2. It is hard to choose which learning technique will produce best result. It is difficult to choose which characteristic contribute more and which contribute less.
3. Computation cost is extremely elevated because similarity matrix is calculated in each step.

A-KNN Approach: Clustering is a technique which is used to assign the elements of similar properties in one cluster and cluster of different properties in another cluster. Clustering is a technique that is used to find out the elements in a data set efficiently. Clustering is an effective in multi dimensionally that is difficult to arrange in effective manner in other environment. The traditional technique of clustering is k-mean clustering. It has disadvantage that it is not easy to identify the initial of k seeds. The main advantage of A-KNN clustering over the HAC is the reduction the amount of the computations. A-KNN algorithms initially consider each entity as a cluster. Each identity is labeled with a unique identifier representing the cluster identity [4]. In the second iteration, assume that K=3, here K is the number of nearest neighbors (NN) to be selected by the user. The algorithm selects the K=3 nearest neighbors to the entity that will be clustered, and then checks labels. When two clusters out of the three clusters have the same label, the algorithm labels the current entity with the same label of those two entities. However, if the three entities have different labels, the algorithm labels the current entity with the same label of the closest entity that is nearest neighbor [4]. The algorithm repeats the clustering process until no more changes occur in the clustering tree. Then the algorithm outputs one cluster at the highest level of the hierarchy. The similarity matrix in AKNN is calculated only once. In adaptive K-mean, the number of elements in each cluster decides the number of comparison for each cluster.

4. PROPOSED METHODOLOGY

A-KNN clustering algorithm is used to cluster the functions of the software for decomposition of software architecture. In A-KNN clustering algorithm, probability of the most relevant function is calculated and using Euclidean distance formula the functions are clustered. In this work, enhanced the Euclidean distance formula to increase the cluster quality. The enhancement based on normalization. In the enhancement two new features will be added. The first point is to calculate normal distance metrics on the basis of normalization. In second point the functions will be clustered on the basis of majority voting. The proposed technique will be implemented in MATLAB. Firstly implement the A-KNN algorithm. Second is enhancement in Euclidean distance formula by using normalization technique. It makes A-KNN algorithm more efficient. A-KNN will properly cluster the highly coupled functions of software. Then implement the proposed technique and compared with previous A-KNN algorithm.
To define the central point of the data on the basis of which data will be clustered is given by equation:

$$\boldsymbol{vi} = \boldsymbol{m}\left(\frac{\mathbf{1}}{\boldsymbol{ci}}\right)\sum_{\boldsymbol{j=1}}^{\boldsymbol{ci}} \boldsymbol{xi} \text{ .. Equation 4.1}$$

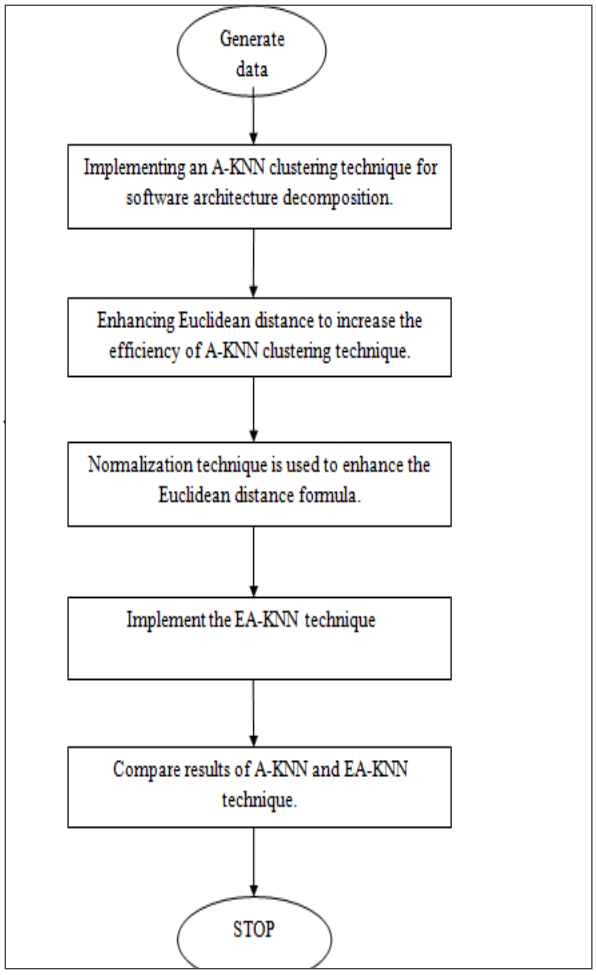


Fig 3: Flow Chart of Proposed Methodology

Pseudo code for proposed Algorithm:

1.Input: Dataset of Software 2.Output: Clustered Data [row column]=size(Dataset);

1. Load dataset and define number of iterations on the dataset
2. Define number of clusters and assign members to each cluster on the basis of uniqueness
3. Define normalization point from the dataset using sigma function
4. Check number of members in each class
If (Class1 members!=Class2 members)
Redefine the position of Normalization point
Else
Assign final position of the normalization point
5. Apply Euclidian distance formula with normalization
6. Plot final results of data clustering

5. EXPERIMENTAL RESULTS

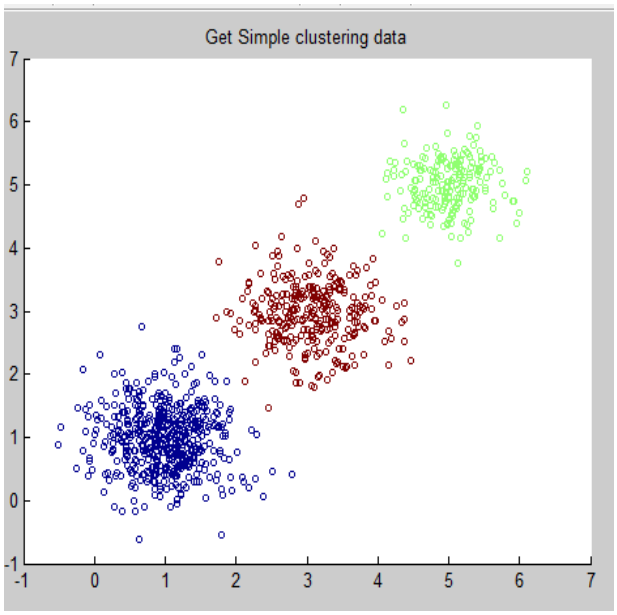


Fig.4: Plotting of Data Points

As illustrated in the figure 4, the A-KNN clustering algorithm is enhanced in which normalization is applied on the RSVP dataset [11] to improve the cluster quality. In the figure the dataset is clustered and showed on the 2D plane with different color.

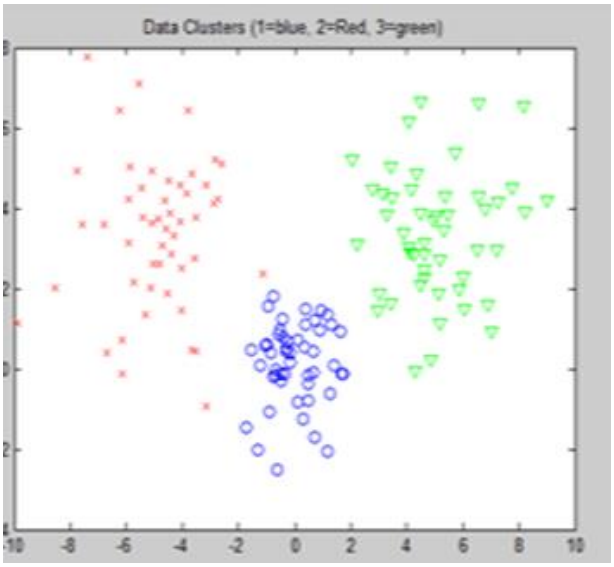


Fig 5: Plotting of data points with different shapes

As illustrated in the figure 5, the uniqueness of each cluster is calculated and data will be scattered according to their similarity on the 2 D plane.

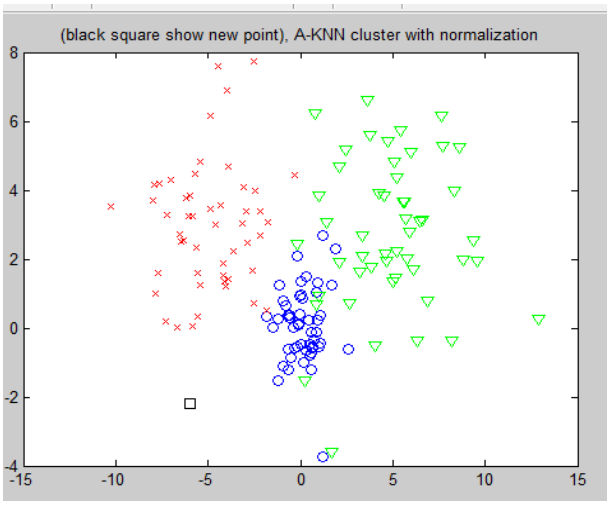


Fig 6: One probabilistic point selected

As illustrated in the figure 6, From the scattered data one point is selected on the basis of probability which is marked with Black Square

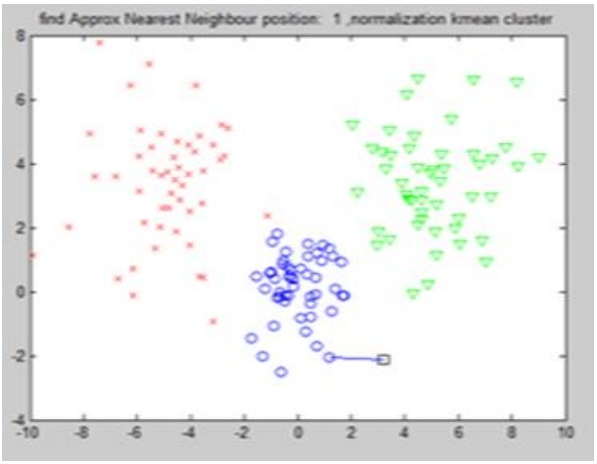
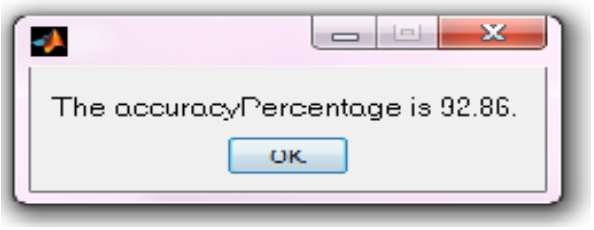


Fig 7: Normalization applied on Euclidean Distance

As illustrated in the figure 7, From the scattered data one point is selected on the basis of probability which is marked with Black Square. After that the distance between probalistic point and data point of cluster is calculated.



As illustrated in the figure 8, the A-KNN clustering algorithm is enhanced in which normalization is applied on the RSVP dataset [10] to improve the cluster quality.

In the previous figure clustered data is shown with three different colors and uniqueness of each cluster is calculated and data will be scattered according to their similarity on the 2D plane. From the scattered data one point is selected on the basis of probability which is marked with Black Square. From that square black point distance to each point in the cluster is calculated. The black point moved to another cluster and distance is calculated to another point. Final position is calculated and on the basis of final point whole data is clustered. Finally, the accuracy of the Enhanced A-KNN clustering algorithm is calculated.

6. CONCLUSION

Software engineering is sequence of steps to produce the software from initial stage to its final stage. Software architecture is important for develop the software. Software Architecture is most important factor for the development of complex and large software system. Architecture decomposition decreases the software complexity. Clustering is creating groups of objects, or clusters, in such a way that objects in the same cluster are very similar and objects in different clusters are quite distinct. Clustering results are helpful to architect or designer for decomposition of architecture. Several clustering techniques are studied. A-KNN clustering technique gives the best result than others clustering technique. In this report, A-KNN algorithm has been used for decomposition of software Architecture with enhancing the Euclidean distance formula based on normalization that is increase the accuracy of A-KNN clustering technique. The proposed enhancement has implemented for decomposition of software architecture focus on functional requirements and attributes. The results demonstrate that enhanced A-KNN is best then previous A-KNN clustering technique. Increase the clustering quality of A-KNN.

7. REFERENCES

1. Shtern, Mark. "Methods for evaluating, selecting and improving software clustering algorithms". Diss. YORK UNIVERSITY TORONTO, 2010.
2. Alkhalid, Abdulaziz, Chung-Horng Lung, and Samuel Ajila. "Software architecture decomposition using adaptive K-nearest neighbor algorithm."Electrical and Computer Engineering (CCECE), 2013 26th Annual IEEE Canadian Conference on. IEEE, 2013.
3. Alkhalid, Abdulaziz, et al. "Software Architecture Decomposition Using Clustering Techniques." Computer Software and Applications Conference (COMPSAC), 2013 IEEE 37th Annual. IEEE, 2013.
4. Lung, Chung-Horng, Marzia Zaman, and Amit Nandi. "Applications of clustering techniques to software partitioning, recovery and restructuring." Journal of Systems and Software 73.2 (2004): 227-244.
5. Alkhalid, Abdulaziz, Mohammad Alshayeb, and Sabri Mahmoud. "Software refactoring at the function level using new Adaptive K-Nearest Neighbor algorithm." Advances in Engineering Software 41.10 (2010): 1160-1178

6. Bozhikova, Violeta. "Using Clustering to Achieve Quality Software Structure."Electronics Technology, 2006. ISSE'06. 29th International Spring Seminar on. IEEE, 2006.
7. Lung, Chung-Horng, Xia Xu, and Marzia Zaman. "Software architecture decomposition using attributes." International Journal of Software Engineering and Knowledge Engineering 17.05 (2007): 599-613.
8. Beck, Fabian, and Stephan Diehl. "Evaluating the impact of software evolution on software clustering." Reverse Engineering (WCRE), 2010 17th Working Conference on. IEEE, 2010.
9. Maqbool, Onaiza, and Haroon Atique Babri. "Hierarchical clustering for software architecture recovery." Software Engineering, IEEE Transactions on 33.11 (2007): 759-780.
10. Awduche, et al. *"RSVP-TE:* extensions to RSVP for LSP tunnels." RFC 3209, December, 2001.
11. Zhang, Lixia, et al. "Resource ReSerVation protocol (RSVP)--version1 functional specification." Resource (1997).