# 7. HYBRID CLIQUE PERCOLATION FOR OVERLAPPING COMMUNITY DETECTION

Community detection problem has been extensively studied and community detection algorithms can be categorized into disjoint and overlapping algorithms. Traditional graph clustering algorithms partitions a graph to disjoint communities such that every node belongs to exactly one cluster. Clique percolation method is the commonly used community detection approach for finding overlapping communities. The implementation of general clique percolation and its variants on a sample network presented in the previous chapter yielded better results but the challenge of missing overlapping communities still persists to a little extent. To solve this issue and produce improved results of community detection further by identifying missing and overlapping communities, a modified clique percolation method is designed. This chapter details on the hybrid clique percolation for overlapping community detection using the k-core communities and z-score.

## 7. 1 INTRODUCTION

In case of graph partitioning, a number of groups and the approximate size of those groups are known in advance and the task is usually to divide the network into the required number of disjoint sub-graphs of the almost same size. But in community detection, the number of communities present in the network and the sizes of the communities are not known in advance Community detection approach assumes that most of the real-world networks divide naturally into groups of nodes or community with dense connections internally and sparser connections between groups. The number and size of the groups are thus determined by the network itself and not by the experimenter.

In many social and information networks, nodes participate in multiple communities i.e., communities tend to overlap. This problem is significantly more complex than the related domain of detecting disjoint communities. In CPM, a typical community is likely to be made up of several cliques that share many of their vertices. A k-clique community is a union of all k-cliques that can be reached from each other through a series of adjacent k-cliques. CPM is devised to extract such k-clique communities of a network such that k-clique communities allow overlaps. It performs an extensive search on the space of cliques, searching for pair of k-cliques that share k-1 nodes. The search space is optimized in an optimized clique percolation method using the global quantity of the given network. The computational time is

reduced by parallelizing the process in parallel clique percolation method. The effectiveness of the clique percolation method in detecting the overlapping communities depends on the size k of the clique which is used to iterate the search process. Hence the optimal k value which is local to the communities is determined to detect all possible overlapping nodes and communities [93].

## 7.2 HYBRID CLIQUE PERCOLATION METHOD

Clique percolation method is a clique based overlapping community detection algorithm works based on the assumption that a community comprises of overlapping sets of fully connected subgraphs. The connection between the nodes within the community is dense such that edges within a community form cliques due to their high density. So this algorithm detects communities by searching for adjacent cliques. It begins by exploring all the k-cliques i.e. cliques of size k in the network. When all the k-cliques have been found a new graph commonly referred to as clique-graph is constructed where each vertex represents a k-clique. Two nodes in this clique graph are connected or adjacent if they share (k + 1) members. Each connected component in the clique-graph represents a community. The clique percolation algorithm has been explained in detail in chapter 6.

If two cliques share $k - 1$ node, then they percolate into each other and are merged into the same community. It merges communities only if they share a larger number of nodes but by making the merge criteria stricter, it gets weak coverage of the network. If k is too high, then it excludes all communities that do not contain a clique of at least size $k$, which is overly strict when $k>7$. One solution is to merge two cliques only if the smaller clique is at least x% embedded in the larger clique. When the threshold is increased, the coverage of the graph is not decreased. With this variation, it is highly essential to specify an optimal clique size [94].

In this work, k-core communities generated by recursively removing all nodes with degree smaller than k from a graph G using maximal k-core algorithm are used to find the optimal clique size for improving the effectiveness of CPM, as the maximal k-core algorithm explained in chapter 5 proved to be efficient in detecting communities than maximal k-clique and maximal k-plex.

**Determining the Optimal Clique Size**

K-core algorithm detected 150 sub communities from given network. The sizes of sub-communities found using k-core follow Gaussian distribution and each size value is standardized using Z-score. Z-score represents the number of standard deviations from the

mean of a sub-community size. The following formula is used to find the Z scores of the k-core sub-community sizes.

$$\boldsymbol{Z = X - \mu/\sigma}$$

where X is the size of each subgraph, 952.46 is the mean of all sizes, 323.12 is the standard deviation. The best optimal Z score is determined as the best optimal value of clique size k for the hybrid CPM.

The main advantage of standard scores is that it always assumes a normal distribution and the scores can be interpreted as a standard proportion of the distribution of the nodes in the communities from which they are calculated.

**Hybrid Clique Percolation Method (HCPM)**

The network G and the clique size k which is chosen the optimal z-score of k-core communities are taken as input for CPM. All k-cliques present in the network G are identified. A new network referred as clique-graph, $G_C$ is formed where each node represents an identified clique and two nodes (clique) in the network, $G_C$ is connected by an edge if they share k + 1 member. Connected components in $G_C$ are identified and then each connected component in $G_C$ represents a community. Thus, the set of communities forms the identified community structure for the network G. The algorithm is able to discover all possible maximum cliques in the network and the number of iterations is increased from k-1 to k+1 to identify every clique in the network.

**Algorithm**

Input: Graph G, clique size k

Output: Overlapping Communities C

Process:

The network, G and the clique size, k

Step 1: G ← Generate graph from twitter

Step 2: Identify all of maximal k-core communities for k=3 in G using kcore (G, k),

Step 3: Dertermine all of maximal k-core sizes using the function X=Size (kcore (G, k)),

Step 3: Caclulate Z-score with mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ of X

Step 4: Find the optimal Z-score value and assign to k

Step 6: Discover overlapping communities using CPM-clique (G, k)

## 7.3 HYBRID OVERLAPPING COMMUNITY DETECTION MODEL

The building blocks of the hybrid overlapping community detection model are i) input component ii) process component iii) output component. The input component deals with data extraction from twitter data of a sports person's network and conversion to network structure. The edge list of network data is used as input. The process component uses a hybrid clique percolation method (HCPM) presented above to find overlapping communities. Initially, the k-core algorithm is used to find sub-communities from the network. Z-score is then applied to the node count of each of these subgroup communities found using the k-core algorithm. The best optimal solution is found and is given as input to the clique size of clique percolation method. The output component generates overlapping sub-communities and their measures with respect to ground truth data. The architecture of the hybrid overlapping community detection model is shown in Fig.7.1.

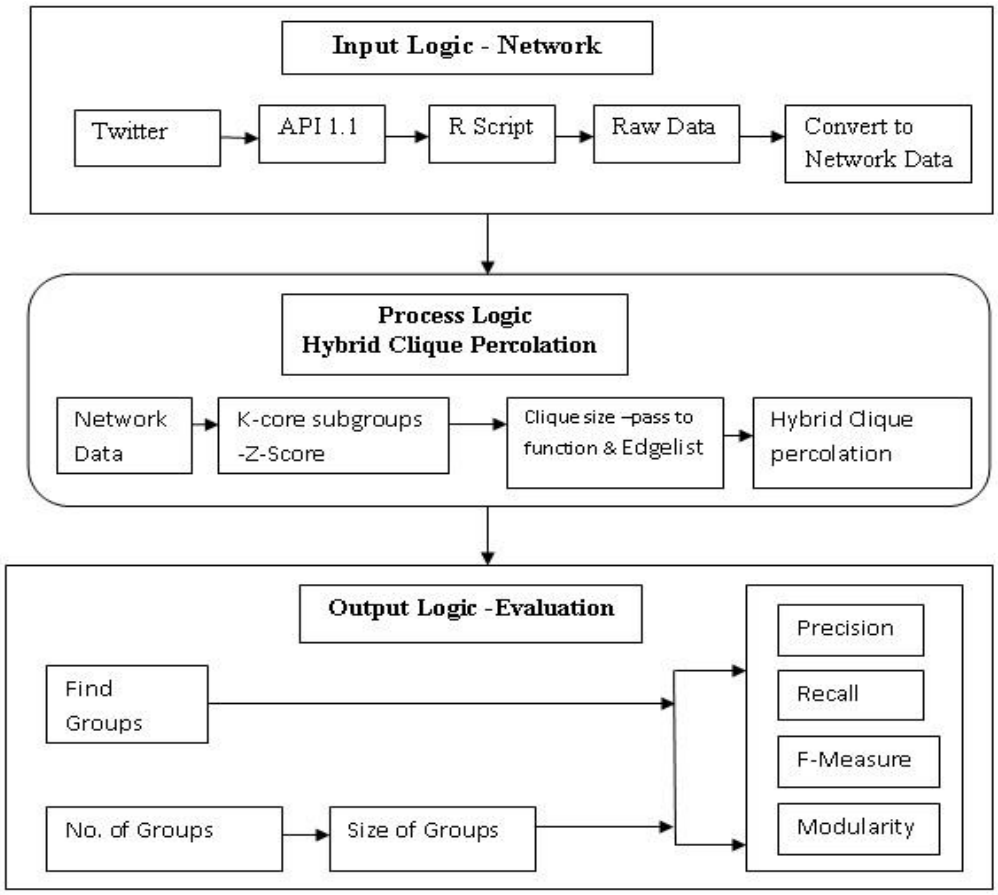


**Fig. 7.1 Proposed Clique Percolation Framework**

## 7.4 EXPERIMENTS AND RESULTS

The implementation of this hybrid approach of overlapping community detection is carried out in R tool. The same twitter network data described in chapter 3 is used for testing the effectiveness of the HCPM. The results of k-core algorithm obtained in chapter 5 are considered initially, which resulted in 150 subgroups communities from a given network. Both small and large subgraph communities are detected and the smallest k-core size is 13 and the highest k-core size is 1684 of the sports person's network. The Z score values of all 150 sizes are computed and the best optimal Z score is found to be 3.28. The Z score values for a sample of 25 community sizes out of 150 communities with their respective sizes are tabulated in Table XXXV.

**Table XXXV k-Core Size of Sub-Groups**

| **Subgroup** | **Size of group (x)** | **$Z=x-\mu/\sigma$** |
|---|---|---|
| Aamir Khan | 1359 | 1.260062 |
| Aaron Finch | 1378 | 1.318885 |
| AlbieMorkel | 1408 | 1.411765 |
| AneeshGautam | 1381 | 1.328173 |
| AshwinRavichandran | 1398 | 1.380805 |
| Cristiano Ronaldo | 1383 | 1.334365 |
| Dale Steyn | 1464 | 1.585139 |
| Gary Kirsten | 1385 | 1.340557 |
| Kevin Pietersen | 1389 | 1.352941 |
| mark boucher | 1390 | 1.356037 |
| Mike Horn | 1366 | 1.281734 |
| Mumbai Indians | 1413 | 1.427245 |
| NehaDhupia | 1391 | 1.359133 |
| R p singh | 1404 | 1.399381 |
| Rahul SharMa | 1415 | 1.433437 |
| Ritika | 1393 | 1.365325 |
| Roger Federer | 1533 | 1.798762 |
| Ross Taylor | 1394 | 1.368421 |
| Salman Khan | 1537 | 1.811146 |
| sonamkalra | 1374 | 1.306502 |
| SonamKapoor | 1419 | 1.44582 |
| Usain St. Leo Bolt | 1600 | 2.006192 |
| Vijay Mallya | 1397 | 1.377709 |
| VVS Laxman | 1356 | 1.250774 |
| YOUWECAN | 2012 | 3.281734 |

***Results of hybrid clique percolation***

The optimal Z-score of k-core community size i.e. 3.28 is selected as the optimal clique size to execute CPM for implementing hybrid clique percolation. HCPM overlapping algorithm discovered 220dense communities from the sports person's network. Out of 220, 115 communities have a large number of nodes with sizes of the subgroups 1800 to 501 and 87 communities have a medium number of nodes with the size of the subgroups 500 to 101 in the community of the network. Twenty-two communities are having a small number of nodes and community sizes range from 20 to 100. The overlapping communities generated by HCPM are shown in Fig.7.2.

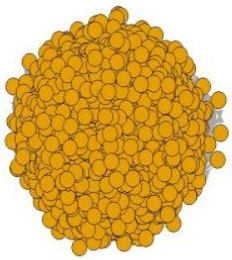

**Fig. 7.2 Communities Detected By HCPM**

A sample of 10 communities detected by HCPM with node ids is given below.

[1] 74 103 1969 1966 1965 1377 726 725 724 722 189 66 92 1817 1814 1803 1191 1190 1188 1187 1186 1185 1184 1180 176 1111

[2] 73 104 1960 1959 1956 1955 1954 1953 1951 1950 1949 1947 1946 1945 1944 1658 1356 1308 887 796 717 1698 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[3] 39 50 1202 1201 1198 1194 1191 190 1188 1187 1186 1185 1184 1180 1176 1111 1698 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[4] 11 55 435 433 430 426 425 422 419 418 417 416 415 414 412 411 410 409 408 407 406 405

[5] 17 18 100 607 606 605 604 597 321 210 411 410 409 408 407 406 410 667 666 664 660 659 658 655 654 652 646

[6] 12 13 14 15 34 35 37 41 45 47 48 49 50 51 5356 57 58 90 91 92 93 94 95 101 102 103 104 106 107 110 121 122 145 166 167 168 169 170

[7] 13 1427 96 250483 486 490 492 493 494526 525 524 518 514 501 499 411 410 409 408 407 406 683 682 680 679 677 676 674

[8] 19 100 667 666 664 660 659 658 655 654 652 6461184 1180 1176 1111 1698

[9] 17 18 607 606 605 604 597 321 210 411 410 409 408 407 406 400 683 682 680 679 677

[10] 14 96 526 525 524 518 514 501 499 411 410 409 408 407 406 40

The time taken by HCPM is 0.17. The modularity score obtained by HCPM is 0.81. The different sizes of the overlapping communities are established for each community in the network. The overlapping community detection method discovered the different size of dense

and sparse communities in the network. Out of 220, there are 172 dense communities and 48 sparse communities detected. The size of the largest community obtained is 1610 and the size of the smallest subgroup is 59. The results for 10 HCPM overlapping communities derived for the given network are presented in Table XXXVI and illustrated in Fig.7.3.

**Table XXXVI Sizes of HCPM Overlapping Communities**

| Overlapping Communities | Size of Communities |
|---|---|
| HCPM 1 | 856 |
| HCPM 2 | 1470 |
| HCPM 3 | 1599 |
| HCPM 4 | 799 |
| HCPM 5 | 1105 |
| HCPM 6 | 1003 |
| HCPM 7 | 1440 |
| HCPM 8 | 1769 |
| HCPM 9 | 884 |
| HCPM 10 | 1320 |

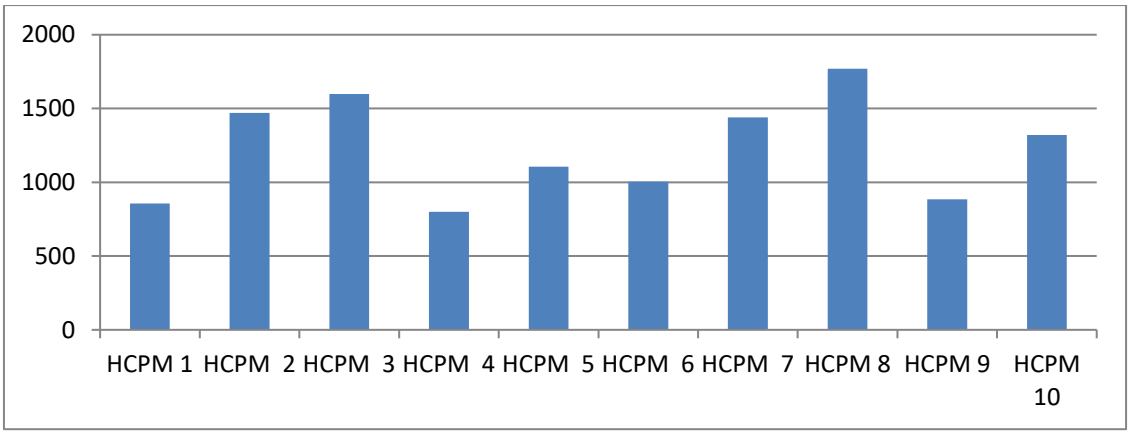


**Fig. 7.3 Sizes of HCPM Overlapping Community**

Also, in-degree of 98 communities lies between 501 to 1800 and the in-degree of 74 communities lies between 101 to 500 which indicate that friends and followers are more interactive with other nodes. The in-degree of 48 communities lies between 20 to 100, which show less interaction with other nodes because it is a very popular node in the network. The high out-degree of 160 communities lies between 101 to 250. High out- degree value of 112 communities suggests more interaction from the outer node to these nodes. For other 60 communities, the out-degree lies in the range of 20 to 60. The degree measures of overlapping communities are evaluated using hybrid clique percolation algorithm and the

results for a sample of 10 communities are presented in Table XXXVII. The in-degree and out-degree of all 220 communities of the sample input network are illustrated in Fig. 7.4.

**Table XXXVII Degree Measures of HCPM Communities**

| HCPM Communities | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| In-degree | 1200 | 961 | 791 | 537 | 964 | 846 | 536 | 834 | 1563 | 583 |
| Out-degree | 76 | 58 | 96 | 130 | 75 | 88 | 98 | 137 | 157 | 89 |

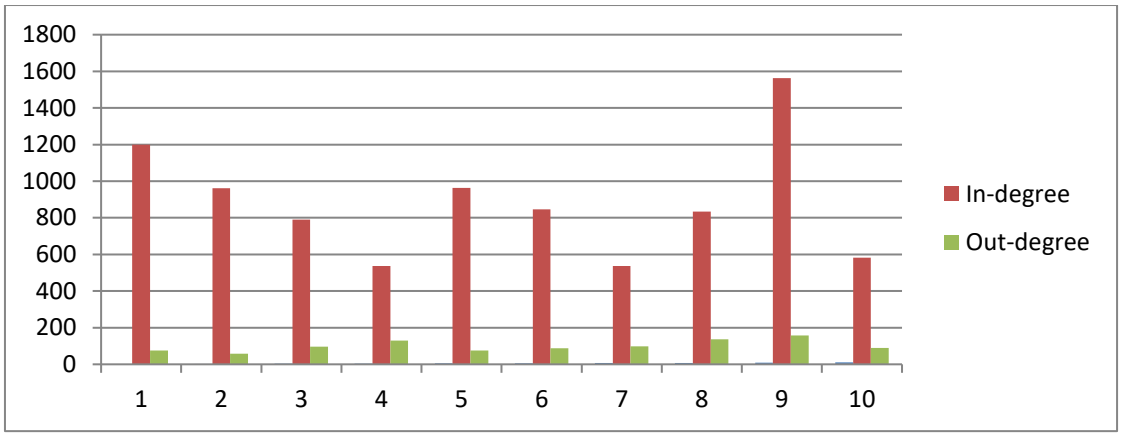


**Fig. 7.4 In-Degree and Out-Degree of Overlapping Communities by HCPM**

The effectiveness of HCPM in identifying missing overlapping communities is determined using measures like precision, recall, F-score by comparing the predicted communities against the ground truth communities of the given network as described in section 6.4. The HCPM yielded the results of precision as 0.79 whereas the recall and the F-measure are 0.69 and 0.78 respectively. The results of various analytical measures are tabulated in Table XXXVIII.

**Table XXXVIII Analytical Measures of HCPM Communities**

| | |
|---|---|
| **Number of Communities** | 220 |
| **Dense Communities** | 172 |
| **Sparse Communities** | 48 |
| **Size of Largest Community** | 1610 |
| **Size of Smallest Community** | 59 |
| **Largest In-Degree** | 1661 |
| **Largest Out-Degree** | 213 |
| **Modularity Score** | 0.81 |
| **F measure** | 0.78 |
| **Precision** | 0.79 |
| **Recall** | 0.69 |

***Comparison of HCPM with CPM, OCPM, PCPM***

The overlapping community detection technique found different sizes of dense and sparse communities in the network. CPM algorithm discovered 198 communities whereas OCPM found 180 communities. HCPM found 220 communities with dense overlapping communities in the network. It shows better performance than other methods because every overlapping community has a large number of nodes in the network. HCPM discovered more number of communities than CPM wherein some communities are sparse and more are overlapping communities. Here networks detected by OCPM have modularity of 0.78 whereas CPM produces 0.77. PCPM exposed 170 communities in the network and the modularity score obtained is 0.846. The modularity of HCPM is almost the same as PCPM and larger than CPM, confirming that there is more interaction between nodes in the network. The comparative results of various critical measures shown by HCPM and other three CPMs are summarized in Table XXXIX.

**Table XXXIX Comparative Results of HCPM with other Three CPMs**

| Method | Number of Communities | Dense Communities | Sparse Communities | Highest In-Degree | Highest Out-degree | Size of the Largest Community | Size of the Smallest Community | Strong Communities | Weak Communities | Modularity Score |
|---|---|---|---|---|---|---|---|---|---|---|
| CPM | 198 | 134 | 64 | 1698 | 204 | 1690 | 49 | 123 | 44 | 0.77 |
| OCPM | 180 | 129 | 51 | 1791 | 214 | 1710 | 51 | 118 | 34 | 0.79 |
| PCPM | 170 | 148 | 22 | 1760 | 210 | 1790 | 53 | 136 | 15 | 0.84 |
| HCPM | 220 | 172 | 48 | 1661 | 213 | 1610 | 59 | 148 | 58 | 0.85 |

The search space problem is optimized by reducing the number of iterations and the computation time is reduced in OCPM. Also, the computation time is reduced by parallelizing the process in PCPM. But HCPM has taken user, system and elapsed time as 84.01, 0.17 and 85.05 respectively. Though the computation time in hybrid approach is little high, the HCPM is effective than CPM, OCPM, and PCPM in identifying the cliques with sharing k+1 nodes as the k value is standardized with k-core sizes. The nodes missed out by three CPMs can be recognized using hybrid clique percolation. Table XXXX and Fig. 7.5 shows the time duration of four different types of clique percolation implementations.

**Table XXXX System Elapsed Time**

| Algorithm | User (seconds) | System (seconds) | Elapsed (seconds) |
|---|---|---|---|
| **CPM** | 83.76 | 0.15 | 84.05 |
| **OCPM** | 37.89 | 0.26 | 38.11 |
| **PCPM** | 0.29 | 0.05 | 0.83 |
| **HCPM** | 84.01 | 0.17 | 85.05 |

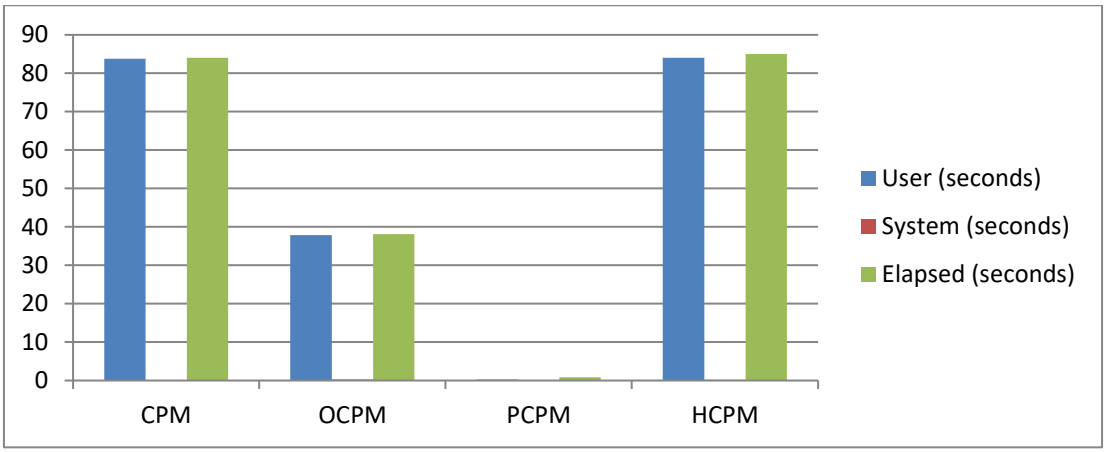


**Fig. 7.5 System Elapsed Time**

The efficiency of HCPM in identifying overlapping communities are compared against CPM based on measures like precision, recall, F-score. These measures are computed by comparing the predicted communities and the ground truth communities of the given network given in chapter 6. The precision, recall, F-measure values obtained for HCPM, OCPM, PCPM, and CPM methods are shown in Table XXXXI and illustrated in Fig. 7.6. HCPM is found to have better precision, recall and F-measure than all the three clique percolation methods in identifying overlapping communities.

**Table XXXXI Quality Measure of HCPM and Three CPMs**

| Measures | CPM | OCPM | PCPM | HCPM |
|---|---|---|---|---|
| **Precision** | 0.76 | 0.78 | 0.79 | 0.79 |
| **Recall** | 0.57 | 0.58 | 0.59 | 0.69 |
| **F measure** | 0.73 | 0.75 | 0.77 | 0.78 |

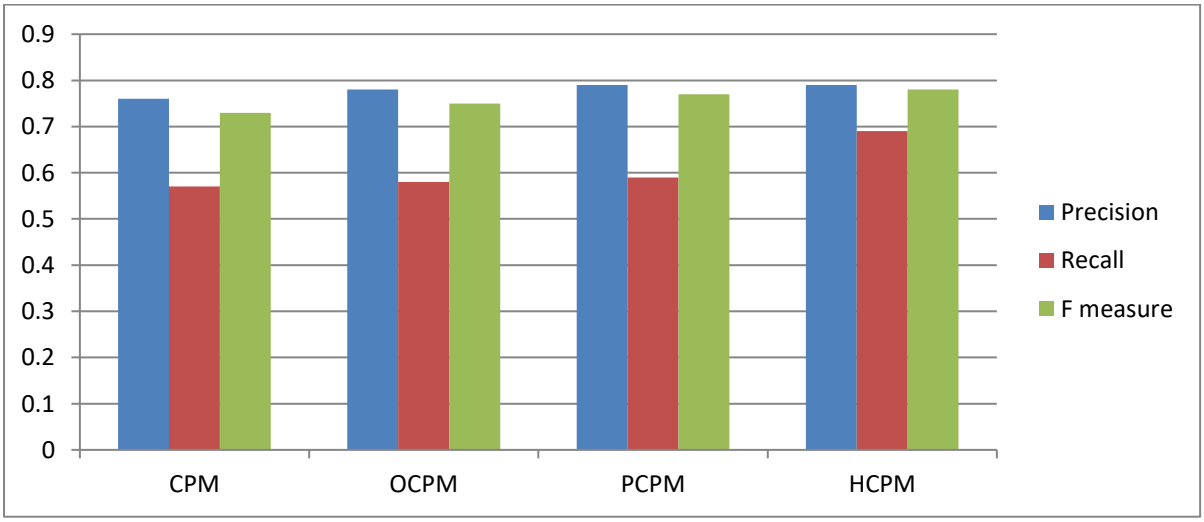


**Fig. 7.6 Comparison of HCPM Quality Measures with CPM, OCPM, PCPM**

***Findings***

From the comparative results, it is found that various analytical measures produced by HCPM are improved than CPM and its variants. The modularity score of communities detected by HCPM is almost the same as that of PCPM and larger than that discovered by CPM and OCPM. The high modularity of HCPM confirms a higher density of communities. The computation time is reduced in PCPM due to parallelization. Though the time taken by HCPM is higher, the HCPM is effective than CPM, OCPM and PCPM in identifying the missing nodes and the cliques with sharing k+1 nodes as the k value are standardized with k-core sizes. The proposed hybrid clique percolation method outperforms in recognizing overlapping communities than CPM, OCPM, PCPM as the evaluation metrics precision, recall, and F-measure are high in HCPM. The empirical result analysis of HCPM algorithm on twitter network data and exhaustive experiments of various overlapping community detection algorithms described in chapter 6 ascertain that the community detection quality has been improved with variants of clique percolation methods.

**SUMMARY**

Clique percolation method of overlapping community detection has been modified using optimal Z – score of k-core communities and its demonstration on sports person's network data has been illuminated in this chapter with results and analysis. The quality of community detection and the effectiveness of hybrid clique percolation method in detecting missing overlapping communities evaluated using ground truth communities is analyzed with various performance metrics. The comparative analyses of HCPM with general CPM and OCPM were also presented with tables and charts in this chapter. Another novel approach to enhance the clique percolation method for overlapping community detection based on association rule mining will be described in the next chapter.