

10-1-2019

Innovative ranking strategy for IPL team formation

Arnabi Mitra

Government College of Engineering and Leather Technology, Govt. of West Bengal

Saptarshi Banerjee

Government College of Engineering and Leather Technology, Govt. of West Bengal

Debayan Ganguly

Government College of Engineering and Leather Technology, Govt. of West Bengal

Ritajit Majumdar

Indian Statistical Institute, Kolkata

Kingshuk Chatterjee

Government College of Engineering & Ceramic Technology

Follow this and additional works at: <https://digitalcommons.isical.ac.in/journal-articles>

Recommended Citation

Mitra, Arnabi; Banerjee, Saptarshi; Ganguly, Debayan; Majumdar, Ritajit; and Chatterjee, Kingshuk, "Innovative ranking strategy for IPL team formation" (2019). *Journal Articles*. 668.
<https://digitalcommons.isical.ac.in/journal-articles/668>

This Research Article is brought to you for free and open access by the Scholarly Publications at ISI Digital Commons. It has been accepted for inclusion in Journal Articles by an authorized administrator of ISI Digital Commons. For more information, please contact ksatpathy@gmail.com.

Innovative Ranking Strategy For IPL Team Formation

Arnabi Mitra, Saptarshi Banerjee, Debayan Ganguly, Ritajit Majumdar, Kingshuk Chatterjee

Abstract: . Indian Premier League (IPL) is a tournament of twenty over cricket matches. Teams of this tournament are selected via an auction from a pool of players. Each team employs a think-tank to build the best possible team. Few studies have been performed to automate the process of team selection. However, those studies mostly concentrate either on the current form of the players, or their long term performance. In this paper, we have (i) selected traditional features as well as determined some derived features, which are generated from the traditional features, for batsmen and bowlers, (ii) formulated heuristics for clustering batsmen into openers, middle order batsmen and finishers, (iii) formulated heuristics for relative ranking of batsmen and bowlers considering the current performance as well as the experience of each player, and (iv) have proposed two greedy algorithms for team selection where the total credit point of the team and the number of players in each cluster is fixed. Our proposed ranking scheme and algorithm not only determines the best possible team, but can also determine the best alternate player if one of the target players is unavailable.

Keywords : Heuristic, Ranking, Greedy Algorithm

I. INTRODUCTION

Cricket is the most popular sports in India. Different for-mat of this game has gained popularity in different times, and in recent times Twenty-twenty (T20) format of the game has gained popularity. In last eleven years, Indian Premier League (IPL) has created a position of its own in the world cricket community. In this tournament, there are eight teams named after eight cities of India. Each team is owned by one or more franchises. A pool of players is created for an auction. Each player is allotted a base price, and the maximum amount each franchise can spend for its entire team is fixed. The auction determines the team for each franchise.

Naturally, the aim of every franchise is to buy the best players from the auction who can help them win the tournament. Each franchise, therefore, maintains a think-tank whose primary job is to determine the players whom they want to buy from the auction. This is not a trivial

task be-cause (i) it is not possible to always pick the best players since the budget for a team is fixed, and (ii) often it so hap-pens that some other franchise buys a player who was in the target list of a franchise. In such situations, the think-tank must determine the best alternate player for the team. The history of IPL has repeatedly seen teams failing to perform in the tournament due to poor player selection. In this paper, we have developed a recommendation sys-tem for player selection based on heuristic ranking of players, and a greedy algorithm for the team selection. The algorithm can help the think-tank to determine the best potential team, and an alternate player if their target player is not available. Previous studies [1, 2, 3, 4] concentrate either on the current form of the players [5, 6], or their long term performance history [7]. However, these two factors individually are not sufficient to decide whether a player is to be bought. Other factors such as, whether a batsman is an opener, or middle-order player or finisher, and consideration of features pertinent to those ordering is of utmost necessity. For

example, middle-order batsman can afford a lower strike rate if he has a good average, but not a finisher. Furthermore, it is necessary to determine a balance between the recent form of a player, and the past history of a good player whose recent form may not be up to the mark. In this paper, we have considered a set of traditional and derived features and have quantified them. Not all of these features are equally important for every player in every position. Therefore, we have broadly classified a potential team into multiple positions, and for each position we have heuristically determined the appropriate weight for these features. For each player in the pool, we have obtained a score based on these weighted features, and have ranked them accordingly. The ranking obtained by this technique is in accordance with the well known ranking of players in IPL. Finally, a relative score on the scale of 1-10 is allotted for each player. Moreover, a fixed basis score is allocated for a team of 15 players, which emulates the fixed budget assigned to each team. We then use greedy algorithm to select the best team within this budget using the aforementioned ranking scheme. The rest of the paper is organized as follows - In Section 2, we define the traditional and derived features which have been considered for batsmen, and quantify them. Three clusters - openers, middle-order and finishers, have been defined in Section 3, each having a heuristic scoring formula which is a weighted sum of those features. The batsmen have been ranked into these clusters according to their points by these heuristics. The features for the bowlers are quantified in Sec-tion 4 and the bowlers are ranked accordingly. In

Revised Manuscript Received October 05, 2019

Arnabi Mitra Department of Computer Science and Engineering, Government College of Engineering and Leather Technology, Kolkata, India,

Saptarshi Banerjee Department of Computer Science and Engineering, Government College of Engineering and Leather Technology, Kolkata, India,

Debayan Ganguly Department of Computer Science and Engineering, Government College of Engineering and Leather Technology, Kolkata, India, Email:

Ritajit Majumdar Advanced Computing & Microelectronics Unit, Indian Statistical Institute, Kolkata, India,

Kingshuk Chatterjee, Department of Computer Science and Engineering, Government College of Engineering and Ceramic Technology, Kolkata, India,

Section 5, we further assign credit points to the players according to their ranks. We present two greedy algorithms for selecting the best IPL team from the previous ranking when the total credit point of the team is fixed. We conclude in Section 6.

II. ANALYZING FEATURES FOR BATSMEN

We have created a database of all the players and their performance in the last eleven seasons of IPL. Some players, who have already retired, are removed from the database. The performance values for those players who have not played some of the early seasons are assigned 0 for those seasons. This comes handy later on while determining the experience factor. For analysis of current form, we have considered the values from the 2018 season of IPL only. In Table 1 we note the traditional features which are considered for the analysis of players. These are very standard features used to report the performance of players in every cricket matches [8], and hence we do not discuss about these. Apart from these features, some derived features are also quantified, which we shall discuss later in this section.

Table 1. Standard features for batsmen and bowlers

Batsmen	Bowlers
Innings	Innings
Runs Scored	Wickets Taken
# Balls Faced	# Balls Bowled
Average	Average
Strike Rate	Strike Rate
# 100s	# Runs Conceded
# 50s	Economy Rate
# 4s Hit	# 4 Wickets
# 6s Hit	# 5 Wickets

We have grouped the batsmen into three position clusters - openers, middle order batsmen and finisher, since these three types of players have three very different role in the match. In the remaining part of this section, we have quantified the features considered in Table 1 for grouping. However, these features conform to all batsmen, and hence are not sufficient for the clustering. Therefore, we also consider some derived features which take into account the specific roles of batsmen in different position clusters.

- **Batting Average:** This feature denotes the average run scored by a batsman per match before getting out.

$$\text{Avg} = \text{Runs}/(\# \text{ Innings} - \# \text{ Not Out})$$

- **Strike Rate:** Strike rate is defined as the average runs scored by a batsman per 100 balls. The higher the strike rate, the more effective a batsman is at scoring runs quickly.

$$\text{SR} = (100 \times \text{Runs})/(\# \text{ Balls})$$

- **Running Between the Wicket:** Though this is a very frequently used term in cricket, there is no proper quantification of this feature. We have quantified it as the number of runs scored per ball in which fours or sixes were not hit.

$$\text{RunWicket} = (\text{Runs} - \# \text{ Fours} \times 4 - \# \text{ Sixes} \times 6) / (\# \text{ Balls} - \# \text{ Fours} - \# \text{ Sixes})$$

- **Hard Hitting:** T20 is a game of runs, and to win it is necessary to score runs quickly. Therefore, apart from quick running, it is necessary to hit many fours and sixes. "Hard Hitter" is a common term in T20 cricket, but it is not quantified. We have quantified this feature as the number of runs scored per ball by hitting four or six.

$$\text{HardHitting} = (\# \text{ Fours} \times 4 + \# \text{ Sixes} \times 6) / \# \text{ Balls}$$

In addition to these, we have defined a COST feature for each of the features. The set of COST features is used to obtain a relative score of an individual with respect to all the IPL players. Let $f(i)$ denote the value of a feature f for the i -th player. If the total number of IPL players is n , then the cost feature for f is defined as $f(i)/(\max\{f(j)\})$. Using this

$$1 \leq i \leq n$$

formula, we have calculated the cost feature for each of the features discussed above.

Experience of a player is an important criteria which should be considered in addition to the above features. Therefore we have defined experience factor (x_{fact}) as

$$x_{\text{fact}}(i) = \text{innings}(i) / (\# \text{ innings in IPL so far})$$

where $\text{innings}(i)$ implies the number of innings the i -th player has played. Define $\text{range}_{x_{\text{fact}}}$ as follows

$$\text{range}_{x_{\text{fact}}} = \max_{1 \leq j \leq n} \{x_{\text{fact}}(j)\} - \min_{1 \leq k \leq n} \{x_{\text{fact}}(k)\}$$

Then the relative experience of a player ($\text{cost}_{x_{\text{fact}}}$) is defined as

$$\text{cost}_{x_{\text{fact}}}(i) = x_{\text{fact}}(i) / \text{range}_{x_{\text{fact}}}$$

The calculation of cost feature and $\text{cost}_{x_{\text{fact}}}$ is similar for bowlers also.

III. CLUSTERING AND RANKING OF BATSMEN

We have clustered the batsmen into three major categories

- (i) opener, (ii) middle order and (iii) finisher. These three types of batsmen are required to play different roles in the match, and hence are expected to have different skills. A total weight of 100 is divided into the features for each batsman. The division of the total weight into features is heuristic so that it models the skill requirements for batsman in different clusters. Furthermore, the ranking of players obtained by such weight distribution conforms with our known player ranking. In the following subsections we discuss the motivations for weight division in each position cluster, and show the top five players according to our ranking scheme.

3.1 Opening batsman

The responsibility of setting up a good foundation for the team's score lies on the openers. The openers get to face the maximum number of balls, and therefore is expected to have a high average. Furthermore, they need to score quickly in the first power play. So a handy strike rate is also a good indicator of the effectiveness of an opening batsman. Both

these features are equally important and are, therefore, assigned the highest weight of 30 each. Furthermore, an opener is expected to stay on the crease for a long time and score big runs. Therefore, we have assigned a weight of 20 to the number of half-centuries (hc) scored by an opener per innings. Often an opener requires some time to set in, and then start hard hitting. During the time, when an opener is still not hitting hard, he should rotate the strikes quickly to keep the score-board moving. However, the necessity of hard hitting cannot be totally ignored during the powerplay. This motivates us to assign a weight of 10 for both running between the wickets and hard hitting.

Based on the choice of feature and weight division, the relative score of the i-th opener (opener(i)) is determined as

$$\text{opener}(i) = \text{cost SR}(i) \times 30 + \text{cost Avg}(i) \times 30 + (\text{hc}(i)/\text{innings}(i)) \times 20 + \text{cost RunWicket}(i) \times 10 + \text{cost HardHitting}(i) \times 10$$

We have used the notation f (i) to denote the value of the feature f for the i-th player considering all the seasons of IPL. Another notation f [i] is used to denote the value of the same feature considering only the last season of IPL. The relative current score of the i-th opener (curr opener[i]) is determined as

$$\text{curr opener}[i] = \text{cost SR}[i] \times 30 + \text{cost Avg}[i] \times 30 + (\text{hc}[i]/\text{innings}[i]) \times 20 + \text{cost RunWicket}[i] \times 10 + \text{cost HardHitting}[i] \times 10$$

Considering the experience factor for each player, the final rank of the i-th opener is calculated as

$$\text{opener rank}(i) = \text{opener}(i) \times \text{cost}_x \text{fact}_x \times (\text{curr opener}[i]/\text{mean opener}) + \text{curr opener}[i]$$

where mean opener is the average score of all the openers. The top five opening batsman from IPL pool of players and their corresponding point derived according to our ranking scheme is shown in Table 2.

Table 2. Top five opening batsmen according to our ranking scheme

Batsman	Points
AB de Villiers	173.5798
MS Dhoni	159.0942
DA Warner	150.113
V Kohli	133.8061
CH Gayle	132.4749

Four out of the five names are indeed the top openers or first down batsmen in IPL. The striking inclusion in this table is MS Dhoni who is almost always a finisher. However, we shall see in the subsequent subsections that the points obtained by Dhoni as a finisher is significantly higher than his points as an opener. That his name appeared in this table simply shows the effectiveness of Dhoni in a T20 match.

3.2 Middle order batsman

The batsmen in these genre need to provide the stability and also must possess the ability to accelerate the scoreboard when chasing a big total. A middle order batsman must be a good runner between the wickets since it becomes difficult to hit big shots during this phase of the match with the fielders spread out.

Furthermore, often when one or both the openers get out quickly, the middle order batsmen must take up to re-responsibility to score big runs. Therefore a decent average is necessary.

The weights for middle order batsmen have been distributed among the features taking the above requirements into consideration. The relative score of the i-th middle order batsman (middle(i)) is determined as

$$\text{middle}(i) = \text{cost SR}(i) \times 20 + \text{cost Avg}(i) \times 30 + (\text{hc}(i)/\text{innings}(i)) \times 10 + \text{cost RunWicket}(i) \times 25 + \text{cost HardHitting}(i) \times 15$$

In accordance with the calculation for openers, the relative current score of the i-th middle order batsman (curr middle[i])

is determined as

$$\text{curr middle}[i] = \text{cost SR}[i] \times 20 + \text{cost Avg}[i] \times 30 + (\text{hc}[i]/\text{innings}[i]) \times 10 + \text{cost RunWicket}[i] \times 25 + \text{cost HardHitting}[i] \times 15$$

Considering the experience factor for each player, the final rank of the i-th middle order batsman is calculated as

$$\text{middle rank}(i) = \text{middle}(i) \times \text{cost}_x \text{fact}_x \times$$

$$(\text{curr middle}[i]/\text{mean middle}) + \text{curr middle}[i]$$

where mean middle is the average score of all the middle order batsmen. Based on the middle rank, we have sorted all the batsmen in descending order of their score. The top five middle order batsmen, according to our scoring scheme is shown in Table 3.

Table 3. Top five middle order batsmen according to our ranking scheme

Batsman	Points
AB de Villiers	183.9566
MS Dhoni	169.7258
DA Warner	163.6285
V Kohli	150.5608
KD Karthik	137.0331

Once again, the names in this ranking do not require any justification. It is worthwhile to note that Dhoni is present in this list also, and his score is slightly higher than his score as an opener. This shows that Dhoni is more effective as a middle order batsman.

3.3 Finisher

Finishers usually have the task of scoring quick runs in the end of the match. Naturally, strike rate and hard hitting are the most important factors for any finisher. It is difficult for a finisher to score big runs regularly since they usually get to play very few overs. Therefore, average score is not considered for these players. Running between the wicket is also an important factor for these batsmen. These players are also expected to remain not out and win the match for the team.

In accordance to the above requirements, we have calculated the relative score of the i-th finisher (finisher(i)) as follows

$$\text{finisher}(i) = \text{cost SR}(i) \times 40 + \text{cost HardHitting}(i) \times 40 + \text{not out}(i) \times 5 + \text{cost RunWicket}(i) \times 15$$

Innovative Ranking Strategy For IPL Team Formation

The current form of the i -th finisher (curr finisher) is calculated considering only the feature scores for last year.

$$\text{curr_finisher}[i] = \text{cost SR}[i] \times 40 + \text{cost HardHitting}[i] \times 40 + \text{not out}[i] \times 5 + \text{cost RunWicket}[i] \times 15$$

Mean finisher is the average score of all the middle order batsmen. Eventually the total score of the i -th finisher, considering the experience factor is calculated as follows.

$$\text{finisher_rank}(i) = \text{finisher}(i) \times \text{cost}_x \text{ fact}_x$$

$$(\text{curr_finisher}[i] / \text{mean_finisher}) + \text{curr_finisher}[i]$$

Based on the score of finisher rank, the top five finishers in IPL are showed in Table 4 which clearly shows that Dhoni should be used as a finisher rather than an opener or middle-order batsman.

Having obtained the score for each player in these three categories, we assign one or more labels (O (Opener), M (Middle Order), F (Finisher)) to the players. The category

Table 4. Top five finishers according to our ranking scheme

Batsman	Points
MS Dhoni	364.3758
DJ Bravo	248.9014
AB de Villiers	223.4076
YK Pathan	215.7580
KD Karthik	214.2518

in which the player has the maximum score is naturally assigned as a label for that player. However, if a player has a higher (or equal) rank in some other category, then that category is also assigned to that player. Such players can be used interchangeably among those categories. For example, Dhoni is assigned only as a finisher since both his rank and his score is higher as a finisher than the other two categories. However, de Villiers has a higher score as a finisher, but a better rank as a middle order or opening batsman. So he can be used interchangeably among these three categories. Similarly, Karthik can be used both as a middle order batsman or as a finisher.

IV. ANALYZING THE FEATURES FOR BOWLERS

Similar to batsmen, we have considered a set of parameters for bowlers and have quantified them. The features which have been considered are as follows -

- **Wicket Per Ball:** It is defined as the number of wickets taken per ball.

$$\text{wicket per ball} = (\# \text{ wickets taken}) / (\# \text{ balls})$$

- **Average:** It denotes the number of runs conceded per wicket taken.

$$\text{Ave} = (\# \text{ runs conceded}) / (\# \text{ wickets taken})$$

- **Economy rate:** Economy rate for a bowler is defined as the number of runs conceded per over bowled.

$$\text{Eco} = (\# \text{ runs conceded}) / (\# \text{ overs bowled}) = (\# \text{ runs conceded} * 6) / (\# \text{ balls})$$

We have not clustered the bowlers into groups. Instead we have considered two parameters for a good bowler into the

same heuristic. A bowler who can take 4 or 5 wickets should be included in the team. However, it is better to take a bowler who can take 1 or 2 wickets per match rather than a bowler who takes 4 or 5 wickets once in a while. Strike rate and consistency has been together quantified for the i -th bowler as

$$\text{bowler}(i) = (4 * \text{four}(i) + 5 * \text{five}(i) + \text{wicket}(i)) * 6 / \text{ball}(i)$$

where $\text{four}(i)$ and $\text{five}(i)$ denote the number of matches where the bowler took 4 and 5 wickets respectively, whereas $\text{wicket}(i)$ denote the total number of wickets taken in matches where the bowler did not take 4 or 5 wickets. Taking the other features into consideration, we divide a total weight of 100 as follows -

$$\text{bowler val}(i) = \text{wicket per ball}(i) * 35 + \text{bowler}(i) * 35 + (1 / \text{Ave}(i)) * 10 + (1 / \text{Eco}(i)) * 10$$

The current form of a bowler (curr bowler val) is calculated similarly considering only the last season's values. The total point of a bowler, considering both current and overall form, is denoted as

$$\text{final bowler}(i) = \text{bowler_val}(i) \times (\text{curr_bowler_val}(i) / \text{mean_bowler}) \times \text{cost}_x \text{ fact}_x + \text{curr_bowler_val}(i)$$

Based on this ranking scheme, we show the top 5 bowlers in Table 5.

Table 5. Top five bowlers according to our ranking scheme

Bowler	Points
A. Tye	335.3772
A. Mishra	296.390
S. Narine	254.321
P. Chawala	223.7809
R. Jadeja	223.283

In the next section we provide the algorithms for selecting a team of 15 players, where the budget is fixed.

V. GREEDY ALGORITHM FOR TEAM SELECTION

In this section, we propose two greedy algorithms for team selection. Each team, containing n players, is partitioned into the following buckets: $B = \{\text{Opener, Middle-order, Finisher, Bowler}\}$, where each bucket $B[i]$ is a set of k_i players such that

$$k_i = n \quad (1)$$

$$i \in B$$

Each team is allotted a value, which emulates the total budget for a team. The number of players k_i in each bucket $B[i]$ is decided by the user, and the unit for each bucket is $\text{unit}(B_i) = (\text{value} * k_i) / 4$.

5.1 Assigning credit points to players

Players in each cluster are further assigned credit points based on their ranking. This helps us to emulate the base price of a player. If a cluster contains c_n players, and it is partitioned into c_p credit point groups, then each group contains c_n / c_p players. The first c_p players are assigned a to the highest credit point group,

the next c_p players to the second highest credit point group and so on. Finally each group is assigned a credit point that decreases as we go down the groups. This step is necessary because the fixed budget of each team has been emulated as a fixed value for each team. The total credit of the team should not exceed the fixed value.

In the example of the following subsection, we have considered four credit groups with valuation 10,9,8 and 7. However, the number of groups, as well as the valuation can be varied according to the team selection criteria.

5.2 First greedy algorithm

In our first algorithm, we consider wicket-keepers as a separate bucket. Therefore, for our first algorithm, Equation 1 is modified as $\sum_{i \in B} k_i + w = n$, where w is the number of wicketkeepers. In Algorithm 1, we show our first algorithm for team selection.

Algorithm 1 Greedy Algorithm 1 for team selection

Input: The pool of players clustered into one or more of the buckets opener, middle-order, finisher and bowler along with their corresponding rank. The total number of players n in a team, the total valuation value of the team, the number of players k_i in each bucket $B[i]$ and the number of wicketkeepers w in the team, such that $\sum_{i \in B} k_i + w = n$.

Output: An optimal team of n players.

```

1: unit ← value/5
2: for all b ∈ {Wicketkeeper, Opener, Middle-order,
  Fin-isher, Bowler} do
3:   ca pb ← unit × kb
4:   minb ← minimum credit point of a player in bucket b
5:   rem ← ca pb
6:   for pos in 1 to kb do
7:     if rem < minb then
8:       while True do
9:         j ← pos
10:        while rem < minb do
11:          j = j-1
12:          if (credit at j)-1 ≥ minb then
13:            credit at j = (credit at j)-1
14:            rem = rem+1
15:          end if
16:          if rem ≥ minb then
17:            break
18:          end if
19:        end while
20:        if rem ≥ minb then
21:          break
22:        end if
23:      end while
24:    end if
25:    Assign the highest credit ≤ rem in pos
26:    rem = rem - assigned credit
27:  end for
28: end for

```

Algorithm 1 assigns the best possible credit for each player position. The total credit of each position is bounded by the value of unit. If the algorithm comes across any position where the remaining unit is less than the minimum credit in the player pool, then it backtracks and reduces the credits assigned in the previous positions till a player is assignable

in the current position. This, being a greedy algorithm, has the risk that it may end up assigning a few players with best rankings along with a few players with very low ranking.

We now produce a team of 15 players using the proposed algorithm. If a total value of 150 or more is assigned to the team, then players of credit 10 can be selected for each position. Furthermore, a very low value can lead to a very poor team. For our example, we have chosen a value of 135, such that the unit is 9. Furthermore, in our example team, we shall have two wicketkeepers, two openers, three middle-order batsman, two finishers and six bowlers.

Since, the capacity of wicketkeeper is 2, and unit is 9, a total credit of 18 can be assigned to the two wicketkeepers. We first assign a point of 10 to the first position, and the remaining 8 points is assigned to the second position. From the rank of players, who are also wicketkeepers, Dhoni is assigned in the first position with 10 points, and S. Samson is assigned to the second position with 8 points. The team of 15 players, as obtained using the Algorithm 1 is shown in Table 6.

Table 6. A team of 15 players, with a total credit point of 135, selected using Algorithm 1

Position	Player	Credit Point
Wicketkeeper	M.S. Dhoni	10
	S. Samson	8
Opener	D. Warner	10
	K.L. Rahul	8
Middle-order	V. Kohli	10
	A.B. de Villiers	10
	F. du Plessis	7
Finisher	D. Bravo	10
	R. Pant	8
Bowlers	A. Tye	10
	A. Mishra	10
	T. Boult	9
	S. Al Hassan	9
	K. Jadav	9
	M. Johnson	7

5.3 Second Greedy Algorithm

In the team selected (Table 6) using Algorithm 1, both the wicketkeepers are finishers. Therefore, the selected team ends up with four finishers. To avoid this scenario, the second greedy algorithm, which is similar to Algorithm 1, keeps an extra restriction that the two wicketkeepers should not belong to the same bucket. By keeping this restriction, the team is exactly similar to that in Table 6, except that instead of S. Samson, we select

Innovative Ranking Strategy For IPL Team Formation

P. Patel as the second wicketkeeper, who is an opener.

This second algorithm can be easily further modified to ensure the cluster of the wicketkeeper. For example, one can impose a restriction such as one of the wicketkeepers must be an opener. Our proposed algorithm is flexible to handle such restrictions. Moreover, using this algorithm along with the the aforementioned ranking, a franchise can easily determine the best alternate player for a position if one of their target player is not available. For example, both Warner and Gayle have credit point 10, but the rank (and point) of Warner is higher than that of Gayle. Therefore, if Warner is already selected by some other team, then he can be replaced with Gayle. If no player of credit 10 is available, then that position can be filled with players of credit 9, and so on.

VI. CONCLUSION

In this paper, we have shown a heuristic method for IPL team selection. For each player, we have considered some traditional and derived features, and have quantified them. We have clustered the players into one or more of the clusters - Opener, Middle-order, Finisher and Bowler according to the score achieved from those features. We have also taken into consideration both the current from and the experience of a player for such ranking. The ranking obtained by our heuristic scheme is in acceptance with the known player rankings in IPL. Finally we have proposed two greedy algorithms to select the best possible team from this ranking when the total credit point and the number of players in each bucket is fixed.

The future scope of this paper is to incorporate two higher level clusters of batting and bowling allrounders. The selection of the team can also include some more flexible buckets where allrounders are given higher preference than batsman and bowlers. A trade-off between inclusion of an allrounder in the team or a batsman or bowler with higher credit point can be studied. Furthermore, the greedy algorithm for team selection has a shortcoming that it may select some high ranking players with some very low ranking ones. A dynamic programming approach may be studied to ensure more or less equal quality players in the team.

REFERENCES

1. Dey, Paramita, Maitreyee Ganguly, and Sarbani Roy. "Net-work centrality based team formation: A case study on T-20 cricket." *Applied computing and informatics* 13, no. 2 (2017): 161-168.
2. Prakash, C. Deep, C. Patvardhan, and Sushobhit Singh. "A new Machine Learning based Deep Performance Index for Ranking IPL T20 Cricketers." *International Journal of Computer Applications* 137, no. 10 (2016): 42-49.
3. Shah, Sricharan, Partha Jyoti Hazarika, and Jiten Hazarika. "A study on performance of cricket players using factor analysis approach." *International Journal of Advanced Research in Computer Science* 8, no. 3 (2017).
4. Sharma, Sujeet Kumar. "A factor analysis approach in performance analysis of T-20 cricket." *Journal of Reliability and Statistical Studies* 6, no. 1 (2013): 69-76.
5. Petersen, Carl, D. B. Pyne, M. J. Portus, and B. Dawson. "Analysis of Twenty/20 cricket performance during the 2008 Indian Premier League." *International Journal of Performance Analysis in Sport* 8, no. 3 (2008): 63-69.
6. Moore, Andrew, J. David Turner, and A. James Johnstone. "A preliminary analysis of team performance in English firstclass Twenty-Twenty (T20) cricket." *International Journal of Performance Analysis in Sport* 12, no. 1 (2012): 188-207.
7. Perera, Harsha, Jack Davis, and Tim B. Swartz. "Optimal lineups in Twenty20 cricket." *Journal of Statistical Computation and Simulation* 86, no. 14 (2016): 2888-2900.
8. Beaudoin, David. *The best batsmen and bowlers in one-daycricket*. Simon Fraser University, 2003.



My name is Arnabi Mitra. I am currently pursuing my BTech in Computer Science and Engineering and Leather Technology from Government College of Engineering and Leather Technology which is in Kolkata, India. I am in fourth year. I have immense interest in the field of Machine Learning and related research work, especially application of Machine Learning in sports domain. This paper is based on the application of Machine Learning in the domain of cricket tournament Indian Premiere League.



My name is Saptarshi Banerjee. I am currently a student at Government College of Engineering and Leather Technology, pursuing my B.Tech in Computer Science and Technology and will be graduating in the year 2020. Being immensely interested and curious on the field of Machine Learning and Data Analysis. I have presented a review paper on Impact of Data Science in Sports in an International Conference and currently completed my internship from Indian Institute of Engineering Science and Technology.



Debayan Ganguly is working as an assistant professor at Computer Science and Engineering department at **Government College of Engineering and Leather Technology**, Plot-II,Block-LB,Sector-III, Kolkata-700106. Before joining this institution he was working as an assistant professor at Computer Science and Engineering department at B P Poddar Institute of Management and Technology,137,VIP Road,Kolkata-70052.He was also worked as a Senior Research Fellow of CSIR ,Government of India working at Indian Statistical Institute, 203 B.T.Road,Kolkata-108.At present Mr. Ganguly is going to complete its Phd in Computer Science and Engineering. His general areas of research interest include computer vision, theoretical aspects of DNA computing and quantum computing. He has several publications in peer-reviewed international journals.



Kingshuk Chatterjee,PhD is working as an assistant professor at Computer Science and Engineering department at **Government College of Engineering and Ceramic Technology**, 73, Abinash Chandra Banerjee Lane, Kolkata-700010.He was a senior research fellow at **Electronics and communication Science Unit at Indian statistical Institute**, 203 B.T.Road,Kolkata-108. His general areas of research interest include computer vision, theoretical aspects of DNA computing and quantum computing. He has several publications in peer-reviewed international journals.



Ritajit Majumdar received his B.Sc. degree in Physics from Asutosh College, University of Calcutta, followed by Post B.Sc. B.Tech. and M.Tech. in Computer Science & Engineering from the University of Calcutta. He has served as an Assitant Professor at B. P. Poddar Institute of Management and Technology for two years. He has been awarded the DST Inspire Scholarship for being the University topper in M.Tech. Presently he is a Junior Research Fellow at Indian Statistical Institute, Kolkata. His research interests include Quantum Error Correction and Quantum Algorithm.