Summary Sheet

Who is the Greatest of All Time (G.O.A.T.)? This has always been a point of contention in the world of sports among athletes, sporting organisations, and fans. This is because different entities have different ways of judging sports figures, often due to inherent bias based on personal preference or nationality, which causes them to gravitate towards their favourite athletes. We may also be affected by recency bias, where we hold newer athletes in higher regard as we can remember their feats better.

Our model introduces an objective and reliable approach to determine which athlete is the greatest based on a single year's performance, as well as the G.O.A.T. of individual or team sports, spanning decades of sporting history.

For Task 1, we made use of the results of the four Grand Slam tournaments in 2018, to determine the greatest female tennis player of 2018. We developed a model based on a weighted directed graph to represent the different matches played by different players, who act as nodes in the graph. In order to find the relative ability of players, we used the Floyd-Warshall Shortest Path algorithm to predict the results of matches between players who did not play against each other. Hence, we can obtain the weighted out-degree to in-degree ratio for an accurate measure of the ability of a player, based on the predicted win-to-loss ratio. Hence, we accurately determined the greatest female tennis player, which agrees with the Women's Tennis Association (WTA)'s 2018 ranking.

For Task 2, we extended our model in Task 1 to include other factors which have to be considered in order to determine the G.O.A.T. of an individual sport. For Task 2a, we chose men's 200-metre butterfly (swimming) to use as an example. As our model uses a weighted directed graph, it can objectively measure the ability of athletes, by comparing the difference in timings within every race. In addition to the relative ability of players, we also incorporated a much wider range of factors, such as the athlete's medal tally, consistency of performance and strength of rivalry with other athletes.

For both Task 1 and 2a, we successfully obtained the greatest player, based on the degree ratio and the G.O.A.T.-ness score defined. The greatest players were easily distinguishable from the rest of the athletes due to their significantly higher scores, which shows that our model is indeed effective in sieving out the best athletes. This also demonstrates that our model is applicable to both one-on-one sports (tennis) and sports with inanimate standards (swimming), and can be easily adapted for any individual sport (Task 2b) by making slight adjustments to the weightage of the different factors used in Task 2a.

Our model in Task 2a can also be modified to find the G.O.A.T. of a team sport (Task 3), by measuring the relative ability of teams, to find the strength of each team. This is to be complemented by the individual contribution of each player. Thus, we can effectively select the best players from the best teams to determine the G.O.A.T. for a team sport.

Contents

I. Letter to Director

Dear Director,

Thank you for your confidence in our team. After conducting extensive testing and research, we would now like to present our model to you. Having tested our model on the men’s 200-metre butterfly to ensure the model's accuracy and reliability, we will demonstrate how our model has successfully determined the G.O.A.T. for men’s 200-metre butterfly.

Our model considers a few key factors that are highly useful for determining the G.O.A.T. of a certain category of individual sports. These are: the athlete's relative ability to other competitors, the athlete's consistency, his or her medal tally, the number of times he or she has broken the world record, and finally, the intensity of rivalry with other competitors. An athlete's relative ability was obtained by modelling the matches between players as interactions with differing strengths.

All of these factors are given a different weightage in the final equation to calculate the “G.O.A.T-ness” of each player, based on their relative importance compared to other factors. For instance, for 200-metre butterfly, we decided that medal count would be given a higher weightage as compared to strength of rivalry, since the athlete's number of medals directly reflects his or her exceptional abilities in the field.

Our model has determined Michael Phelps to be the G.O.A.T. for this event, with his "G.O.A.T.-ness" score being significantly larger than the next best swimmer. This is highly reasonable given his outstanding performance, impressive medal tally, and the fact that many news organisations recognised him as the G.O.A.T. As such, our model is able to objectively and conclusively identify the G.O.A.T. of various sports.

Furthermore, our model not only considered factors that are featured in various world athlete rankings, but also rivalry and consistency, which carry great importance when determining the G.O.A.T. This ensures that the G.O.A.T. identified is not a newcomer or a rising star, but instead one who has been consistently outperforming other athletes in the sport.

We hope that you have gained a better understanding of how our model works, and that you will strongly consider adopting our model to determine the G.O.A.T. for various categories of sports in the future. If you have any further queries about our model, please feel free to contact us, and we will be very pleased to share more details with you.

Regards,
Team XX

1. Introduction

The term "G.O.A.T." - the *Greatest of All Time* - has been popularised in recent years, with ardent sports fans constantly debating which professional athlete deserves such a prestigious title. Yet, there is a wide range of factors that make it difficult for sports commentators to reach a consensus on who the G.O.A.T. is, including the athletes' positions on world sports rankings, number of matches won, and other noteworthy achievements such as world records. Thus, in this study, we aim to develop a model that determines the G.O.A.T. for different individual and team sports as objectively as possible, based on competition performance and other unique achievements.

Current athlete rankings for various sports, ranging from individual games such as chess to team sports such as soccer, include a set of common factors. For instance, world rankings by World Athletics are based on the measured results of athletes and their placing during competitions, with extra points given to athletes who achieve new world records [1].

Also, the Fédération Internationale de Football Association (FIFA) uses a ranking system where each team has its own rating point. Points are evaluated based on several factors, including the strength of opponent (which is based on the opponent's ranking), the match importance (based on the type of competition), and the outcome. There is also different weightage given for scores attained at different timings; the weightage of more recent scores is higher [2].

Our work thus takes into account the more relevant factors mentioned above, and splits them into "objective performance factors" and "subjective perception factors". Thus, our model can be applied more realistically to various sports, which may depend heavily on one type of factor more than another.

Therefore, to determine the G.O.A.T. of various sports, we developed an algorithm that is easily adaptable to suit the nature of results for many sports. In addition, we have conducted test cases to assess the accuracy, viability and sensitivity of our model.

2. Restatement of Problem

Top Sport, a sports network, has requested our team to develop a model for individual sports, and use it to determine the G.O.A.T. of one individual sport of our choice.

This problem requires us to complete 4 tasks:

1. Determine the greatest woman tennis player of 2018 based on the results of the four Grand Slam tournaments in 2018, using a model.
2. Determine the G.O.A.T. of any individual sport of our choice using a model, and discuss the changes required for this model to determine the G.O.A.T. of other individual sports.
3. Discuss the changes required for the model from (2) to determine the G.O.A.T. of team sports.

4. Write a letter to the Director of Top Sport to describe our model and the G.O.A.T determined in (2).

3. Task 1

3.1. Assumptions

In order to simplify the model and make it possible to be computed mathematically, assumptions would have to be made. These are the key assumptions that we made, and their justifications:

1. Walkover matches are not factored into the model.
   Justification: Walkover matches are not an indication of one's ability of reputation, as they are mostly a result of injuries and other similar reasons against the player's will [3].

2. The tennis player's match performance is an accurate representation of the player's tennis abilities. The players' are equally well-rested before each match.
   Justification: The most significant determinant of a player's performance in each match is likely to be his or her skills/abilities, and other conditions of the athletes, such as fatigue and emotional stress, are difficult to account for because such factors cannot be easily quantified.

3. The home-field advantage, which refers to the tendency for sports performers to win more often when competing at their home facility, of the female tennis players is low and negligible.
   Justification: A research study found that although some degree of home advantage exists for men's tennis, the performance of female tennis players appears to be unaffected by home advantage [4]. In addition, the impact of home advantage is difficult to quantify, as the extent of home advantage can vary across different athletes. As we intended our model to be a more objective form of assessment, we excluded this factor from our model.

3.2. Variables

To compare the abilities of the female tennis players in the four Grand Slam tournaments in 2018, we create a function d_u to evaluate the performance of these players within the scope of the four tournaments given. This function would take into account the following variables:

Variable`	**Definition**
u_i, v_i	The two players involved in match i
$m_{u,v}$	Total number of matches played between player u and player v
s_{u_i}	Score of player u for match i
$g(u, v)$	Difference in the total score of player u and player v for match i i.e. $\sum s_{u_i} - \sum s_{v_i}$; if $g(u_i, v_i)$ is positive, this represents a directed edge from u to v and vice versa
$h(u, v)$	Average winning margin between player u and player v based on number of matches

Variable`	Definition
d_u	Degree ratio $\frac{\sum_{i=1}^{n} h(u,i)}{\sum_{j=1}^{m} h(j,u)}$ i.e. the ratio of the sum of all the weights of outward edges to the sum of all the weights of inward edges

3.3. Model Development

Factors affecting the greatness of a tennis player

These are the factors we have considered to determine the greatness of a tennis player:

- **The margin that a player wins another player by**
  By winning another player by a larger margin, it indicates that the winner has a much greater skill level relative to her opponent, as the winner can defeat her opponent with greater ease.
- **The number of players that a player wins**
  If a player can win more opponents, her ability should be ranked higher than others as she is more capable. In addition, winning more players indicates that she is likely to have progressed to later stages in the tournaments, such as semi-finals or finals, and this is an indicator of greatness.

We did not make use of the seed rankings provided, as that would defeat the purposes of our model, since it would already introduce some bias in our model beforehand. This is to ensure greater objectivity and that our rankings are solely based on the results of the 2018 tournaments.

Assessing skill of player using weighted directed graph

When determining the greatest female tennis player of 2018, it is critical to assess the performance of the athlete with respect to other athletes competing in the same sport. Hence, we modelled the matches played between different players as a directed graph of nodes and directed edges. Each node represents a player, and each directed edge is drawn from the more proficient player to the less proficient player. This is determined by the polarity of $g(u_i, v_i)$, which is given by the following equation:

$$g(u,v) = \sum s_{u_i} - \sum s_{v_i}$$

$g(u,v)$ represents the **total winning margin between player u and player** v, which can be represented by the difference between $\sum s_{u_i}$, the total sum of player u's scores in the matches played between player u and v, and $\sum s_{v_i}$, the total sum of scores of player v in these same matches. A positive value will indicate that player u has a positive net margin over player v, which indicates that player u has a higher skill level than player v.

To compare this relative skill level with other players, from $g(u,v)$, we averaged out the margins based on $m_{u,v}$, the **number of matches played between player u and** v, as shown by the following equation:

$$h(u,v) = \frac{g(u,v)}{m_{u,v}}$$

$h(u,v)$ will form the edge weight in our directed network. All edges will have positive edge weight, and hence the direction will represent the polarity of $h(u,v)$. In other words, if $h(u,v)$ is positive, the edge will be directed from player u to player v. If $h(u,v)$ is negative, the edge will be directed from player v to player u.

By averaging $g(u,v)$, we ensure that the number of matches played has little effect on our edge weight, as this will be purely a gauge of relative ability of one player compared to another, in order to determine who is better, and how much better. Hence if a player plays many matches with another player, and wins many of them, she will not be disproportionately rewarded in our model, especially if the other player is not as skilful and they play many matches with each other.

Filling in missing information using the Floyd-Warshall Algorithm

As players do not play with all other players in the tournament, it can be difficult to determine the relative skill level of players if they do not play a match. In order to fill in the missing information, we used the Floyd-Warshall algorithm, which is an All-Pairs Shortest Path (APSP) algorithm that finds the shortest path in a directed weighted graph between all possible pairs of nodes (i.e. players) in the graph.

The Floyd-Warshall algorithm can be used to add edges (or relationships) between players even if players have not played with each other before. It assumes the worst-case scenario, to determine the minimum margin player u wins player v by. If this minimum margin is higher than the minimum margin player v wins player u by, then player u is more likely to win player v, and vice versa. This can be determined using the Floyd-Warshall shortest path algorithm on the directed graph to determine the shortest path between two players u and v, by taking edge distance to be the weight of the directed edge, given by $h(u,v)$. If no directed edge exists from player u to player v, the edge weight is set to zero so that player u never wins player v in any scenario.

Using the new edge weights obtained from the Floyd-Warshall algorithm, we reconstruct all the possible edges between all players, in a new directed graph. In this new directed graph, the edges are directed from the winning player to the losing player, which includes our predictions for players who have never played with each other before. This allows us to predict the margins between such pairs of players in the 4 Grand Slam tournaments, hence compensating for the lack of data and greatly increasing our accuracy in finding the player who is most consistently outperforming others, as we can determine the proportion of players a particular athlete can win more accurately.

For each player u, we can then calculate the **degree ratio** d_u, which is the weighted ratio of out-edges against in-edges of the directed graph. This is an indicator of the probability of winning, which accounts for both the frequency that the player wins at, as well as the margin that the player wins by, which are both essential factors when gauging the ability of an athlete. The degree ratio is given by the following formula:

$$d_u = \frac{1 + \sum_{i=1}^{n} h(u,i)}{1 + \sum_{j=1}^{m} h(j,u)}$$

A constant of 1 was added to the numerator and denominator so that d_u is always defined.

The weighted degree ratio is an accurate indicator of ability, because it represents the predicted win-to-loss ratio of each player, which is a gauge of a player's ability to defeat other players in the sport.

3.4. Results

We ranked the tennis players based on their calculated degree ratio, d_u. A higher degree ratio is reflective of a more skilled tennis player, because she has a higher total probability of winning against other players, and a lower total probability of losing against other players.

Actual WTA Ranking (2018)	Ranking based on our model	Name of tennis player	Degree ratio
1	1	Simona Halep	204.83
2	2	Angelique Kerber	83.67
5	3	Naomi Osaka	62.07

Table 1: Our ranking of the top 3 female tennis players based on degree ratio

As shown in Table 1, we found that the greatest female tennis player of 2018 was Simona Halep. Also, our ranking of the top 3 female tennis players in 2018 closely matches that of the actual Women's Tennis Association (WTA) rankings [5]. This shows that the accuracy of our model is high, and is an objective method of analysing the players' abilities solely based on 2018 results, instead of judging the players based on past performance. In fact, Naomi Osaka, who won her first ever Grand Slam title in the 2018 US Open [6], was featured highly in our rankings, showing that our results are indeed determined on the basis of 2018 results.

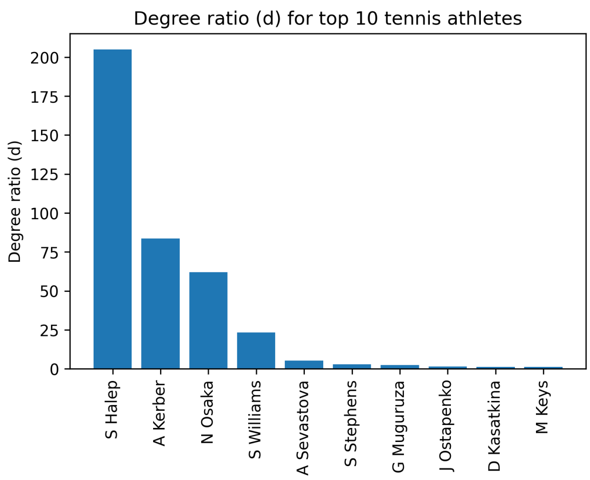


Fig. 1: Degree ratio for top 10 athletes in women's singles tennis in 2018

As shown in Fig. 1 above, our results are deterministic as there is an extremely large difference in the value of d_i between Simona Halep, and the rest of the players. We can thus conclusively determine, without much doubt, that she is the greatest player for 2018 women's singles tennis.

However, some discrepancies still arise because the WTA takes into account the points earned at every tournament during a 52-week stretch, while our model only takes into account the results of the four Grand Slam tournaments.

Also, the number of points awarded for each tournament are determined by how far players advance, and thus accounts for preliminary rounds, such as the round-of-128 and round-of-64, while our model only looks at the results for the round-of-16 and onwards.

Fig. 2 below shows the graph obtained from the raw data of the matches between different players. The red node represents Simona Halep, the best female tennis player in 2018 according to both our model's rankings and WTA's 2018 rankings. The green nodes represent the players that Simona Halep won in the four Grand Slam tournaments, such as Sloane Stephens and Angelique Kerber. The directed edge is drawn from the winning player to the losing player, and the thickness of the edge is an indicator of edge weight. The network representation of our model is consistent with our results, which suggests that Simona Halep is consistently outperforming many of the most skilled players in singles women’s tennis by significant margins.

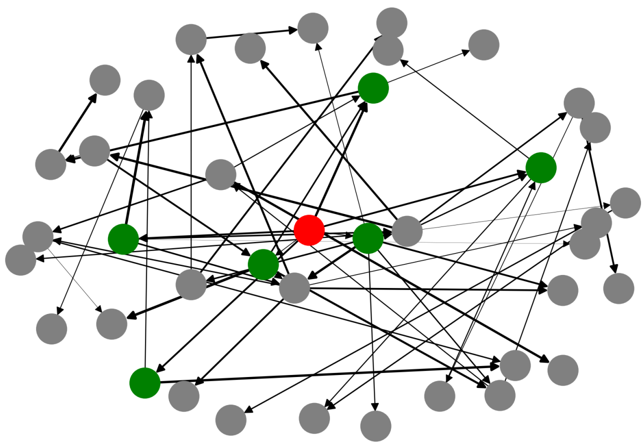


Fig. 2: Weighted directed graph of athletes competing in women’s singles tennis in 2018 before running the Floyd-Warshall algorithm

3.5. Strengths of Model

Our network representation is advantageous as it can **represent many more relationships between different players** as compared to a simple hierarchical scoring method. By averaging our edge weights over the number of matches, we can obtain a fairly accurate understanding of the relative abilities between players.

Via the Floyd-Warshall algorithm, our model also extrapolates the data to estimate the probability of each tennis player winning against all other tennis players. This is because each player does not get the chance to play against every other player, thus our model offers a more definitive ranking of the players, as compared to only using the match results between players who have competed against each other before. If we had not filled up the missing edges, we may be imposing an unfair penalty on athletes who lost in the semifinals and finals. These athletes are skilled, but they lost to other more skilled competitors. Hence, the Floyd-Warshall algorithm is essential to **ensure fairness** in our evaluation of the relative ability of athletes.

Compared to heuristic models, which includes more randomness, our model is **deterministic**, which means that it always executes in a similar fashion and produces the same answer. This greatly increases the credibility and validity of our model, such that *Top Sport* will be more inclined to adopt it.

Our results are able to conclusively determine the greatest player, in this case Simona Halep, with a clear distinction from the rest of the players using our method. This indicates that Simona Halep is undoubtedly the greatest female single's tennis player of 2018.

3.6. Limitations of Model

However, it should be noted that the Floyd-Warshall algorithm is **computationally intensive**, as it has a time complexity of $O(N^3)$. where N is the total number of tennis players in the data set. Hence, while this algorithm may have been feasible for this data set, for sports with a much larger number of players (e.g. more than 1000 distinct players), this limitation may be more pronounced.

4. Task 2

4.1. Chosen Sport (Task 2a)

For this task, the sport that we chose is the men's 200-metre butterfly event in swimming. This choice was made due to the differences to the mode of competition for tennis. This is because tennis is a "one-on-one" match-based competition between two players, allowing edges to be easily drawn between players based on match results, whereas swimmers compete based on an inanimate standard (the fastest swimmer wins), adding an additional layer of difficulty and complexity for our model, and demonstrating the vast applicability of our algorithm.

We have selected two prestigious international competitions to obtain data from, the Olympic Games [7] and FINA World Championships [8], over a time period of 1990 to 2020, so as to compare a large number of athletes from all nationalities and across several decades. We chose these two competitions as they have the most complete sets of data over the specified time period, and include the results from both finals and semifinals, allowing us to make use of a larger range of data.

4.2. Additional Assumptions

On top of the assumptions made in Section 3.1 that the matches are accurate representations of ability, and there is negligible home field advantage, we need to make further assumptions:

- Swimmers who have been embroiled in severe scandals (e.g. use of performance-enhancing drugs) are not worthy of the G.O.A.T. title due to poor morals, and are thus excluded from consideration in our model.
  Justification: Athletes involved in scandals often gain large negative press and have a poor reputation [9] as they are viewed to have violated the principles of sportsmanship and integrity, thus the public will very likely not support these athletes. Furthermore, many athletes have been banned from competing when caught using drugs, as doping is prohibited by most international sports organisations, including the International Olympic Committee [10].

- We also assumed that the G.O.A.T. will be found within our dataset.
  Justification: The Olympic Games and FINA World Championships are two of the most prestigious international sporting events in the world, hence only the best swimmers from around the world will be able to stand a chance to compete in these two competitions, since countries will only send their best athletes to compete.

- The sports technology used by athletes competing against each other has negligible impact on the relative performance of athletes in the same match. An example of technology used in sports include better swimwear to reduce water resistance. The component of our model to assess athlete ability only draws relations between athletes who have competed against each other, hence we accounted for the improvements in sports technology over time.
  Justification: While it is true that sports technology can give certain athletes an edge over others, and does not accurately represent the ability of athletes, this effect is negligible as countries that athletes are representing often invest in their athletes and provide them with the best possible technology so that they stand the best chance of winning. Therefore the effect on differing levels of equipment and technology is negligible.

4.3. Additional Factors

Given the need to consider a much larger database of results from a variety of competitions, we will introduce additional variables, which can be considered "subjective perception factors", to build on to our previous model, as shown below:

- **Prestige of competition** - Usually, sporting competitions are categorised according to the level and significance of the competition, and the scores achieved in different competitions are given different weightage [11]. The highest category reflects the strongest competitions and consequently awards the most points. For example, the weightage given to the Olympic Games are higher than those at other local competitions.
  According to the categories determined by World Athletics [12], the Olympic Games and various World Championships are in the same category, which shows that they are likely to be equally prestigious. Thus, in our model, we will be according the results attained at the Olympic games and FINA World Championships the same weightage.

- **Special achievements such as World Records -** Athletes who attain a new world record are likely to receive greater publicity than just winning first place, allowing them to gain more recognition in the sport scene. Also, witnessing world records being broken brings great pleasure for athletics fans, thus these athletes would leave a greater impression in the public's eyes. Therefore, such once-in-a-lifetime achievements should be taken into account when determining the G.O.A.T. of various sports, and various sports world rankings such as World Athletics do so too - "bonus points are given as an extra reward for the obvious significance and promotional value of such performance" [13].

- **Medal tally -** The number of medals an athlete receives over time, the more public recognition the athlete receives, as winning first, second or third place across multiple years is a clear signal of these athletes' superior skills compared to other athletes. Thus, the accumulated medal tally should contribute to how likely the public would perceive the athlete as a G.O.A.T. Furthermore, prize-giving ceremonies for various sporting competitions are often heavily publicised, thus helping the athletes to gain fame and respect as well.

- **Consistency of performance -** Athletes who are able to maintain or even improve their skills over time would receive more media hype and public attention for various competitions, as they are consistently viewed as the most likely to win. Furthermore, being able to sustain their performance is testament to their commitment and dedication to the sport, as well as their perseverance and undying spirit, which are hallmarks of a G.O.A.T. Hence, tracking the consistency of athletes' performance across multiple competitions is crucial.

- **Famous rivalry -** Famous rivalries are critical when it comes to swimming, and make swimming matches and the athletes memorable. Rivalries often gain a lot of publicity, and are heavily advertised in the media. Hence, when athletes compete with their rivals, and win the race (often by a fraction of a second), they are considered to be the greatest, especially in comparison to rivals who are already one of the best in the sport. Furthermore, according to the social identity theory, sports fans seek membership in groups that will positively reflect on their self and public image [14], thus public support for both athletes in a strong rivalry will increase greatly, as compared to other athletes.

Variable / Function	Definition
G_i	Measure of "G.O.A.T.-ness"
d_i	Measure of athlete's ability, i.e. weighted degree ratio
b_i	Weighted medal tally based on gold, silver and bronze medals
r_i	Number of times the world record was broken by the athlete
c_i	Measure of consistency

Variable / Function	Definition
$\beta_{i,j}$	Average margin of victory between an unordered pair of 2 athletes
α_i	Rivalry score
$norm(x_i, y, z)$	Normalisation function for variable x_i in range $[y, z]$ This is given by $y + ((z - y) * \frac{x_i - min(x_i)}{max(x_i) - min(x_i)})$

4.4. Model Development

Our overall equation for calculating the "G.O.A.T.-ness" of an athlete is as follows:

$$G_i = d_i \times norm(c_i, 1, 1.5) \times norm(\sqrt{b_i}, 1, 2) \times norm(\alpha_i, 1, 1.25) \times (\sqrt{r_i} + 1)$$

Measuring the ability of an athlete

The ability of an athlete is part of what makes an athlete great. In order to stand out from others, an athlete not only has to do well individually, but has to outperform others to be labelled as great. Hence, our model takes into account one's relative performance to others.

d_i represents the **weighted degree ratio**, which corresponds to the **ability score** of each swimmer. This was calculated using a similar method as Section 3.3 (Task 1). For each race every year, we used the directed graph as explained in Section 3.3. We used data from the semi-finals and finals of each competition, and each race has 8 athletes. For earlier years, the finals had 16 athletes, so we only included the top 8 for each round for fairness. For each race, based on the timings, we constructed a directed edge from athlete u to athlete v if athlete u had a better timing than athlete v during the race. The edge weight is defined as the average of all the margins. In this context, margins refer to the difference in timings for the men's 200-metre butterfly event. This method has been elaborated on in Section 3.3.

Using the model in Section 3.3, we can calculate the weighted degree ratio d_i. Note that the Floyd-Warshall algorithm was not used, as the graph generated was much denser as compared to the graph for women's tennis in Task 1, since swimming is a sport with an inanimate standard and not "one-on-one" sport like tennis. A dense graph refers to a graph where number of edges in a graph is close to the maximum number of edges in a fully connected graph [15]. Hence, there was no need to fill up missing relationships in the graph.

In addition, unlike Task 1, where all the matches occurred in 2018, the competitions in Task 2a spanned over a longer period of time, and it would be unfair to directly compare different athletes from different eras based on their raw timings, without accounting for general improvements over the years. Hence, by not using the Floyd-Warshall algorithm, our model only compares each athlete with other athletes in the same era for fairness of comparison.

Calculating weighted medal tally

Medals are an important form of recognition for an athlete's achievement. Thus, for **medal tally** b_i, we will calculate the total number of "medal points" each swimmer had earned from both the FINA World Championships and the Olympic Games, from 1990 to 2020. Using the existing "sum-ranking system" for a weighted medal tally [16], a gold medal is worth 3 "points", a silver is worth 2 "points", and a bronze is worth 1 "point". This value is then square-rooted to reflect the diminishing returns from each additional "medal point".

Then, for these values to be multiplied to the overall function to obtain a measure of "G.O.A.T.-ness", they will be normalised to a scale of 1 to 2. Thus, an athlete's normalised medal tally represents how close or far he is from the best and the worst athlete, in terms of medal tally. A normalised value of 1 represents the worst medal tally in the data set, and will not affect the value of G_i when multiplied to it, while 2 represents the best medal tally, and will rather significantly amplify the value of G_i.

Number of world records broken

The number of world records broken is an important factor, as it usually makes the news, allowing the athlete to gain international attention. Thus, it is often an athlete's pathway to fame. For the number of world records broken, we chose to represent this variable as $\sqrt{r_i} + 1$, where r_i is the **number of times the world record for men's 200-metre butterfly was broken** by the athlete.

If $r_i = 0$, $\sqrt{r_i} + 1 = 1$, thus the value of G_i will not be affected. However, if $r_i \geqslant 1$, the value of G_i will increase at a decreasing rate, to reflect the decreasing significance of each additional time an athlete breaks the world record. When the athlete breaks the world record the first time, he or she is already regarded as having exceptional skills, and subsequent world records will add to this public image, but with less significance, due to the pre-conceived expectation that the athlete is already extremely talented.

Measurement of consistency

Consistency is critical to ensure that the athlete is truly the Greatest of *All Time*, and not just a one-time phenomenon. The **measure of consistency**, c_i, is based on the number of competitions, or specifically finals, that the athlete had participated in. This indicates that the athlete has been participating in the sporting event for a long period of time, and has sufficient experience in competitions to consistently enter the finals, so that he or she can be highly regarded as one of the greatest athletes of "*all time*", and not just the greatest athlete for a short period of time.

This value, c_i, is then normalised between 1 to 1.5. The scale of normalisation is not as large as that for medal tally b_i due to the interdependency of our chosen factors, as entering the finals more times would naturally translate into a higher chance of attaining more medals. Thus, a small aspect of consistency would have already been accounted for in b_i, the normalised weighted medal count.

Measure of strength of rivalry

$\beta_{i,j}$ between two athletes is defined as the **average margin** (i.e. difference in timing) between them when they are in the same race. A narrow win would be much more significant in the eyes of the public, since it implies that both athletes are similar in ability and there is a possibility the athlete who lost would be able to win in the future. Hence, the smaller the margin, the higher the rivalry score. We considered the average victory margin between medallists. For each athlete, we identified his main rival by finding the opponent for which he has the smallest winning margin. For athlete i with opponents of index 1 to n, the rivalry score of athlete i, α_i, is given by:

$$\alpha_i = max(0, 1 - 2\beta_{i,1}, 1 - 2\beta_{i,2}, \ldots, 1 - 2\beta_{i,n})$$

$$= max(0, 1 - 2\beta_{i,j}) \ \ \forall j \in [1, n], j \in \mathbb{Z}^+$$

Therefore, the rivalry score α_i is given by the strength of rivalry with athlete i's strongest rival. We acknowledge that this is a minor factor as compared to other factors we have previously highlighted in this study, as the athletes' performance is more important, hence a lower normalised weight is placed on the rivalry score (from 1 to 1.25).

The strength of rivalry between 2 athletes decreases as the average win margin increases. We proposed a linear relationship between the rivalry score α_i of an athlete, and $\beta_{i,j}$, the minimum margin the athlete defeats his strongest rival by. When the margin is equal to or above 0.5s, the score is set to 0, as it will be a clear win in that case without any significant rivalry. Since the maximum of the rivalry score α_i is 1, and α_i is 0 when $\beta_{i,j}$ is 0.5, we can represent this relationship using the expression $1 - 2\beta_{i,j}$. The maximum rivalry, α_i, is the maximum of all $1 - 2\beta_{i,j}$.

4.5. Results

Firstly, to analyse our values obtained for the weighted, normalised medal tally, these are the different number of medals, and values for b_i and $norm(\sqrt{b_i}, 1, 2)$ that we obtained for the more notable swimmers, from the period of 1990 to 2020 and in the 2 competitions for men's 200-metre butterfly:

	No. of gold medals	**No. of silver medals**	**No. of bronze medals**	b_i	$\sqrt{b_i}$	$norm(\sqrt{b_i}, 1, 2)$
Michael Phelps	8	1	0	26	5.10	2.0
Melvin Stewart	2	0	0	6	2.45	1.48
Kristóf Milák	1	0	0	3	1.73	1.34

Table 2: Raw data and derived data for top 3 athletes

As shown in the table above, as Michael Phelps has the highest medal tally among all the swimmers, his value of $norm(\sqrt{b_i},1,2)$ is the highest, at 2.0. Those without any medals would have a value of 1.0.

Using our model, these are the values of the different variables that we have obtained, for the athletes with highest G.O.A.T.-ness scores, G_i. Among the top 10 athletes we identified, 4 of them (Michael Phelps, Melvin Stewart, Chad Le Clos, Michael Gross) were ranked among top 6 medallists in the Olympic Games' website [7], thus showing our model's accuracy:

	d_i	$norm(c_i,1,1.5)$	$norm(\sqrt{b_i},1,2)$	$\sqrt{r_i}+1$	$norm(\alpha_i,1,1.25)$	G_i
Michael Phelps	77.30	1.5	2.0	3	1.23	855.76
Melvin Stewart	43.45	1.1	1.48	2	1.0	141.51
Kristóf Milák	31.83	1.05	1.34	2	1.0	89.54

Table 3: Variables for computation of G_i

As shown above in Table 3, our model has determined that Michael Phelps is the G.O.A.T. of men's 200-metre butterfly, as he has the highest value for G_i, and this is not surprising given his exceptional performance in most of the variables that we have selected. As he had taken part in 10 finals from 1990 to 2020, the most for any swimmer in this event, his normalised measure of consistency is the highest. Furthermore, having broken the world record for the men's 200-metre butterfly for a total of 4 times in these competitions ($r_i = 4$), his value of $\sqrt{r_i}+1$ is also the highest.

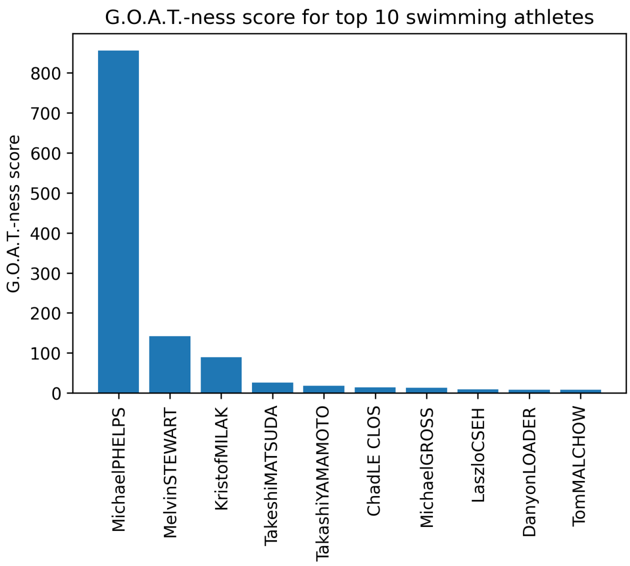


Fig. 3: G.O.A.T.-ness score for top 10 male swimmers in 200-metre butterfly

As shown in Fig. 3, we also observe that the results of our model show similar patterns as Task 1, where the G.O.A.T. has significantly greater ability score d_i, and also significantly greater G_i than the other athletes, which indicates that our model is deterministic as elaborated on earlier. Hence, our model allows us to identify the G.O.A.T. with high certainty. The network of swimmers can be found in the Appendix.

Indeed, several news organisations, including FOX Sports, USA Today Sports, and The New York Times [17, 18, 19], have recognised Michael Phelps as the G.O.A.T. of swimming, or one of the best athletes of all time, due to his "longevity" in the field, and his high medal count and number of world records.

4.6. Strengths and Limitations of Model

Strengths

Our model takes into account several quantifiable factors and splits them into 2 categories - firstly, the differences between the swimmers' timings which allow us to have an **objective analysis of their relative abilities**, and secondly, factors that would shape one's perception of the swimmers, such as their consistency and number of world records. This is crucial for us to gain a **comprehensive overview** of the swimmers' achievements and abilities over a long period of time. Besides including factors that are featured in various global rankings of athletes, which demonstrates our model's relevance to real life, our model also considers time-based factors, such as consistency, which carry great importance when determining the G.O.A.T. This ensures that the G.O.A.T. identified is not a newcomer or a rising star, but instead, one that has been consistently outperforming other athletes in the sport.

Our model also takes into account that the ability of players improve over time [20]. This can be due to advancements in sports science such as better swimwear, more calibrated diets of athletes, and also specially tailored training regimes, leading to a general improvement in the quality and abilities of athletes. In order to factor this into our model, we ensured that edges were only drawn between athletes in the same competition, so that the abilities are compared on a relative basis instead of using an absolute scale.

Also, our model is **flexible** and can be easily adapted for the analysis of different types of sports, which includes both one-on-one sports like tennis, as well as sports with an inanimate standard such as swimming, by adjusting our directed graph model.

Furthermore, our model made use of **real-life data** from the Olympics Games and FIFA World Championships, over a long period of time, which helped to increase the accuracy of our model to truly determine the G.O.A.T.

Lastly, our adapted model that does not use the Floyd-Warshall algorithm takes $O(m)$ time, where m is the number of competition records, thus it is not computationally intensive. Hence, it can be used for calculating large data sets in a **relatively short amount of time**.

Limitations

However, there are still certain factors that we could not account for in our model, such as the swimmers' level of sportsmanship, or other character traits that are not revealed through their performance in competitions. Despite this, these factors **cannot be quantified via mathematical means**, thus this is an unavoidable limitation of our model.

Also, our data was not as comprehensive as we would have liked it to be, as we felt that it was also important to consider smaller-scale or regional competitions, which swimmers would also frequently take part in. However, many of these less-publicised competitions **do not publish sufficient data online** that dates back to 1990, and thus, solely including the recent results of these competitions would make it highly unfair for older swimmers who competed more frequently in the past. Also, including the 2 most prominent and international competitions would already allow us to gain data from the most well-known and skilled swimmers, who definitely have the highest likelihood of being known as the G.O.A.T.

4.7. Sensitivity Analysis

We conducted a few variations of sensitivity analysis on our model. Firstly, we varied the normalisation ranges. In our model, the ranges were from [1,1.25] (α_i), [1,1.5] (c_i) and [1,2] (b_i). We varied the upper bound of the normalisation from 1 to 2, while fixing the lower bound at 1, such that the range of normalisation for each of the three factors varies from 1 to 2. The top 3 athletes remained consistent. The graph below in Fig. 4 shows the ratio of top 2 G.O.A.T.-ness scores as we varied the normalisation of each range independently.

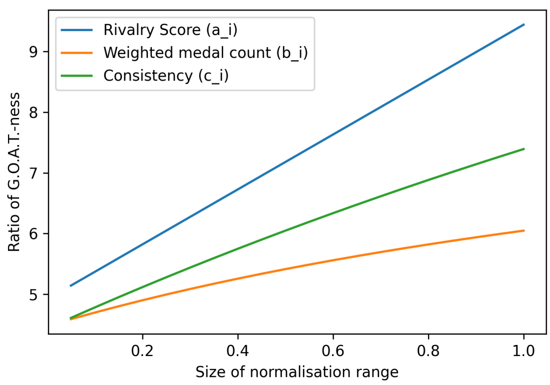


Fig. 4: Variation of ratio of G.O.A.T.-ness score by changing normalisation ranges of different variables

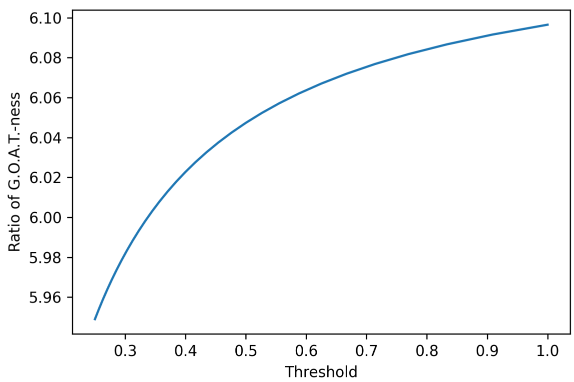


Fig. 5: Variation of ratio of G.O.A.T.-ness score by varying threshold

To test the stability of our results, for each round, we removed 10% of the competition records randomly from the entire dataset, and then found the top 3 greatest swimmers based on their G.O.A.T.-ness score. After executing this process 1000 times, Table 4 summarises the top athletes and their probability of achieving each of the top 3 positions.

Athlete name	1st place	2nd place	3rd place
Michael Phelps	99.9%	0.1%	0%
Melvin Stewart	0.1%	85.1%	4.5%
Kristóf Milák	0%	13.3%	75.8%
Michael Gross	0%	1.1%	8%
Chad Le Clos	0%	0.4%	7.3%

Table 4: Probability of some top athletes achieving each of the top 3 positions, when 10% of data is randomly removed

As seen from the table above, we attain Michael Phelps, Melvin Stewart and Kristóf Milák as our top 3 athletes consistently. Michael Phelps remains as the G.O.A.T. in almost all cases. This shows that our model is robust against changes in the competition data and various parameters used in calculating the G.O.A.T.-ness score.

For the rivalry score, we also varied our threshold for a close margin, which we previously chose to be 0.5s. This threshold is varied between 0.25s to 1.0s. Fig. 5 above shows the variation in ratio of the G.O.A.T.-ness scores of the top two athletes with changing threshold, which shows the ratio of G.O.A.T.-ness scores start to stabilise after the threshold reaches 0.5s. Hence we used 0.5s as our threshold as there would be marginal changes to this ratio after 0.5s.

4.8. Extension of Model to Other Individual Sports (Task 2b)

Due to the differences between one-on-one sports and inanimate sports, it is crucial that we adopt different approaches when attempting to find the G.O.A.T. for the respective sports.

"One-on-one" Sports

“One-on-one” sports are those involving two parties competing directly with one another. Examples of such sports include badminton and tennis (as seen in Task 1). For such sports, we can use a model similar to our model in Task 1, in which the scores of a match between two players are used to create a network with directed edges that allows for comparisons to be drawn between players, and thus a G.O.A.T. can be determined.

Additional factors, such as consistency and medal tally, would have to be added to the model in Task 1, just as we have demonstrated in Task 2. However, the weightage of each factor in the overall function for G_i would have to be different for various sports. For instance, the number of world records should not be factored into G_i, simply because there are no world record timings or scores to surpass.

Furthermore, the component of rivalry can be given greater weightage in one-to-one sports, as the presence of rivalry in one-on-one sports is likely to be higher than sports with inanimate standards.

The nature of these sports allows famous pairs to directly battle against each other, especially in the finals, thus gaining greater attention from the public, and breeding classic rivalries such as Ali and Frasier in boxing.

Sports with an inanimate standard

These sports refer to sports which use rankings, score or time measures to assess athletes' performance; hence, there is a greater emphasis on individual performance. Examples of such sports include golf and swimming (as shown in Task 2a). In such scenarios, we can use a similar model as that used in 2a. As we acknowledge that the abilities of athletes can collectively improve or worsen over time, directed edges can be drawn from players of higher to lower ability based on their differences in score (such as timing or goals), rather than merely using the absolute scores.

Also, edges are only drawn between athletes who had competed in the same competition (i.e. the same round of finals), instead of drawing edges across different competitions, since players from different eras may not be directly competing against each other. We can then derive the ability of athletes from this graph representation as illustrated in Task 2a. Using similar factors in Task 2a, we can find the greatness score, G_i, of each athlete.

5. Task 3

5.1. Additional Considerations

For team sports, there are additional considerations that would have to be made. Team sports often involve a division of labour - each player must undertake a specific role or position, and a set of functions based on the position he or she plays in [21].

For instance, this division is clearly visible in soccer, where players can be the goalkeeper, defenders, midfielders or forwards. Being the main attackers, the forwards are the most likely to score points for the team. However, this would mean that forwards are more likely to be the G.O.A.T., as compared to other roles, should the ranking system depend on the points scored by the player for the team. This suggests that considering the athletes' differing positions could play a key role when assessing performance in team sports.

While it could be possible that the attacker is the most likely to be remembered by the general public and is usually the one who makes headlines, and is thus the most likely to be the G.O.A.T., we still want to recognise exceptional performance in other roles such as goalkeepers.

5.2. Adaptation of Our Previous Model

As such, our measure of "G.O.A.T.-ness" for each athlete in team sports will comprise of 2 components - a "team performance" element and "individual performance" element.

For the **"team performance" element** G_t, the method of calculation will be largely similar to that in Task 2:

$$G_t = d_t \times norm(c_t,1,1.5) \times norm(\sqrt{b_t},1,2) \times norm(a_t,1,1.25)$$

To obtain the value of d_t, which represents **team ability**, we will construct a network where each team represents a node, and the edges drawn between nodes represent the margins that each team wins another team by, thus using the same concept as the previous tasks. In other words, instead of comparing the abilities of individual athletes, we compare the abilities of entire teams.

The **measure of consistency**, c_t, represents how many times the team has successfully entered the quarter-finals or any prestigious equivalent for each sport. b_t is the **total number of "medal points"** that the team has earned over time, and is again square-rooted and normalised from a range of 1 to 2. Lastly, the **degree of rivalry**, a_t, is determined by the number of times the team has competed in the same finals as another team.

It is important to note that the degree of rivalry will again vary from sport to sport, given the different frequency of competition, and "star power" of individual athletes [22]. For instance, the degree of rivalry for American football is higher than other team sports such as basketball and baseball [23]. Thus, the normalisation range for α_i should be different for various sports too.

To account for the fact that athletes could take part in different teams across their sporting career [24], we can take a weighted average of team abilities, based on the duration the athlete has played in each particular team.

As for the **"individual performance" element**, G_{indiv}, it takes into account the different Key Performance Indicators (KPIs) that athletes with different roles in team sports will have. For instance, using the same example of soccer, a study found that the goalkeepers had a vastly different set of KPIs from the outfield players [25]. Hence, the "individual performance" element for goalkeepers could compare the number of balls successfully defended, while that for attackers could look at the total number of successful passes or goals.

Of course, the KPIs across different sports would vary, but quantifiable indicators or metrics should be adopted. There are many existing KPIs for athletes from a variety of sports, based on whether the sports are net and wall games, invasion games, or striking and fielding games [26], and these can be used to quantify the individual greatness of each athlete. In fact, for baseball, there is even an existing metric, known as the Wins Above Replacement (WAR), that assesses each player's contribution to a team's success, and is based on the quality of his or her batting, base-running, fielding, and pitching [27].

Thus, the overall formula for "G.O.A.T.-ness" is as follows:

$$G_{total} = G_t \times G_{indiv}$$

5.3. Strengths and Limitations of this Adaptation

Strengths

Our model for team sports is largely adapted from the one we used for individual sports, hence it is easier for sports networks, like *Top Sports,* to utilise. Since teams are competing against each other, our method for comparing the relative ability of players using the directed graph was easily adapted for teams.

Also, as we recognised the important fact that players may play for multiple teams in their lifetime, we used a weighted average for the ability of teams to ensure fairness.

Next, it is important to consider the individual achievements or performance of athletes even if they play as a team, as we acknowledge that some roles in team sports are uniquely different from other roles, as we have seen in the soccer example. These players are limited to certain areas of the court, or are only involved in defence. This extends to other sports as well, such as baseball and hockey.

Furthermore, as most teams in team sports from basketball to volleyball have substitutes, it is also necessary to consider the contributions of each player, to judge whether he or she is an important player in the team and how much of the team's success can be attributed to him or her.

Nonetheless, in team sports, a good player is one who is able to work well with the team and bring the team to victory [28]. Hence, a model for team sports would also have to take into account the results that his or her team produces in competitions to determine his or her ranking. To summarise, the team scoring system will effectively select the best teams from the rest, and the individual component is critical in differentiating the best players in the best teams.

Limitations

While our model has many strengths, the effectiveness of our model is still dependent on the amount and type of data present. For other sports besides the ones we studied, the type of data and the specific factors involved will vary slightly, which will mean that minor adjustments will still have to be made to our model, to tailor to each specific sports category's factors.

We also did not consider home advantage of athletes, as it was difficult to quantify the impact of home advantage, where athletes have the tendency to perform better when in their home country.

Lastly, we did not take into account the number of exceptional feats, such as world records, that each athlete achieved in team sports, as it is very difficult to objectively determine what is an "exceptional" feat, and individual athletes do not typically achieve world records in team sports. World records usually only occur in individual sports with inanimate standards.

6. Conclusion

In conclusion, in order to determine the G.O.A.T. for individual and team sports, we crafted a metric for "G.O.A.T.-ness", which took into account the relative abilities of players, their medal tallies, number of times they broke the world record, consistency over the years, as well the intensity of rivalry. These are all important factors to consider, given that there is an objective aspect of the player's abilities, as well as subjective factors that affect the public's recognition of different athletes. In addition, to test our model using real-life data, we obtained competition results from two international swimming competitions over the span of 30 years.

In task 1, our model found that the greatest female single's tennis player of 2018 was Simona Halep, and for task 2, we determined that Michael Phelps was the G.O.A.T. of men's 200-metre butterfly. Having compared our results with real-life rankings and various news articles, we found that the different versions of our model are able to accurately and reliably determine the greatest player in a particular year and the G.O.A.T. for individual sports. From our sensitivity analysis, we have also established that our model is extremely stable, and can cope relatively well with missing data or varying range of normalisations.

As for team sports, we have proposed several changes to the "G.O.A.T.-ness" metric, such as taking into account both team and individual performance, so that athletes with different positions can be fairly judged. Different KPIs and normalisation ranges should also be used for different sports.

The diagram below summarises how our initial model for Task 1 (women's tennis) had been gradually adapted for determining the G.O.A.T. for Task 2 (men's 200-metre butterfly), and subsequently for other individual sports, and finally for team sports.

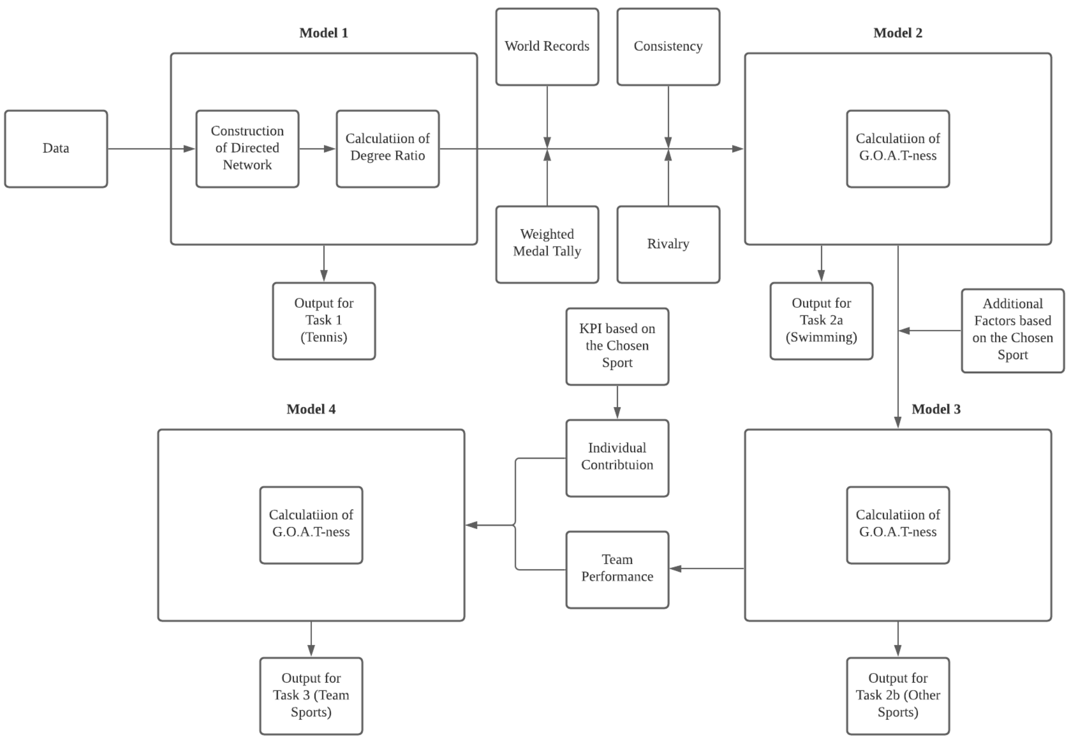


Fig. 6: Summary of how our model was adapted for different types of sports

Lastly, in future studies, with more time, results from a larger range of international competitions can be consolidated, in order to obtain more comprehensive data of the performance of athletes.

7. Bibliography

[1] World Athletics. World Athletics: 1. basics of the rankings. https://www.worldathletics.org/world-ranking-rules/basics.

[2] Xiong, S., Yang, L., Zin, N. A. M., & Iida, H. (2016, November). Mathematical model of ranking accuracy and popularity promotion. In *2016 3rd international conference on systems and informatics (ICSAI)* (pp. 350-357). IEEE.

[3] Howard, M. M. (2017, November 21). The Difference Between a Walkover & a Default in Tennis. Healthy Living. https://healthyliving.azcentral.com/difference-between-walkover-default-tennis-2676.html.

[4] Koning, R. H. (2011). Home advantage in professional tennis. *Journal of Sports Sciences*, 29(1), 19-27.

[5] WTA Top 10 and their best 2018 result. Women's Tennis Association. (2018, November 5). https://www.wtatennis.com/photos/1407642/wta-top-10-and-their-best-2018-result.

[6] Four for Osaka: How Naomi's Grand Slams were won. CNA. (2021, February 20). https://www.channelnewsasia.com/news/sport/naomi-osaka-tennis-four-grand-slams-australian-us-open-14243448.

[7] Swimming - 200m butterfly men. International Olympic Committee. (2020, November 9). https://www.olympic.org/swimming/200m-butterfly-men.

[8] Official FINA Website. FINA. https://www.fina.org/results.

[9] Sato, S., Ko, Y. J., Chang, Y., & Kay, M. (2019). How does the negative impact of an athlete's reputational crisis spill over to endorsed and competing brands? The moderating effects of consumer knowledge. *Communication & Sport*, 7(3), 385-409.

[10] Support and protect clean athletes. International Olympic Committee. (2020, November 9). https://www.olympic.org/fight-against-doping.

[11] FIFA world ranking: How it is calculated and what it is used for. FIFA world ranking: How it is calculated and what it is used for | Goal.com. https://www.goal.com/en-us/news/fifa-world-ranking-how-it-is-calculated-what-it-is-used-for/16w60sntgv7x61a6q08b7ooi0p.

[12] World Athletics. World Athletics: 6. Overall Rankings. https://www.worldathletics.org/world-ranking-rules/overall-rankings.

[13] World Athletics: 2. Track & Field Events. worldathletics.org. https://www.worldathletics.org/world-ranking-rules/track-field-events.

[14] C. T. (2018). The impact of the phenomenon of sport rivalry on fans. *The Future of Fandom*, 28.

[15] Dense graph. dense graph. (2019, December 2). https://xlinux.nist.gov/dads/HTML/densegraph.html.

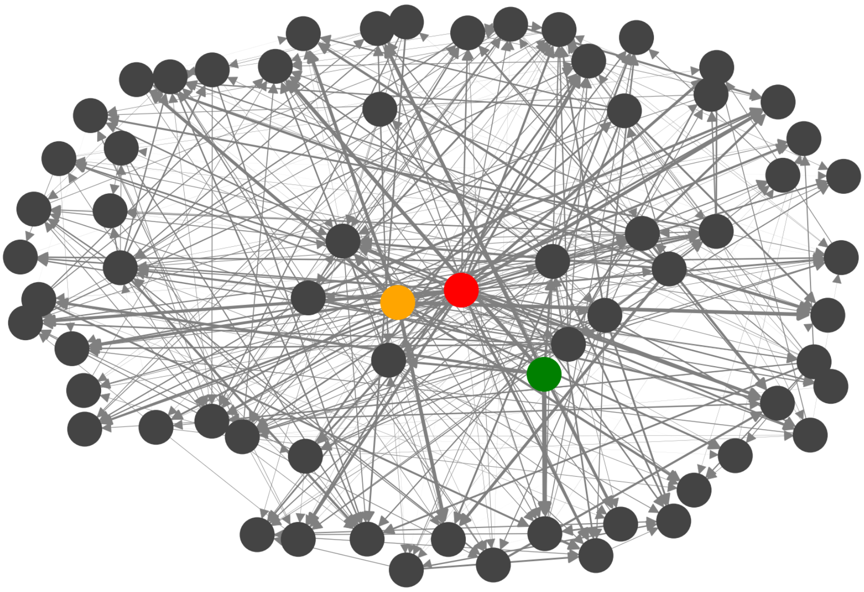
[16] Du, J. (2018). Modifying Olympics Medal Table via a Stochastic Multicriteria Acceptability Analysis. *Mathematical Problems in Engineering*, 2018.

[17] Chase, C. (2016, November 15). Michael Phelps or Usain Bolt: Who's the Olympic G.O.A.T.? FOX Sports. https://www.foxsports.com/stories/olympics/michael-phelps-or-usain-bolt-whos-the-olympic-g-o-a-t.

[18] Martinelli, M. R. (2019, August 6). The Olympic race Michael Phelps had no business winning proves why he's the GOAT. USA Today. https://ftw.usatoday.com/2019/08/michael-phelps-goat-swimming-olympics-medal-count-record-2008-butterfly.
[19] The New York Times. (2012, August 2). The Greatest Olympian. The New York Times. https://www.nytimes.com/2012/08/02/opinion/michael-phelps-goat.html.
[20] Costa, M. J., Marinho, D. A., Reis, V. M., Silva, A. J., Marques, M. C., Bragada, J. A., & Barbosa, T. M. (2010). Tracking the performance of world-ranked swimmers. *Journal of sports science & medicine*, 9(3), 411.
[21] Arastey, G. M. (2019, November 22). Performance Indicators in Football. Sport Performance Analysis. https://www.sportperformanceanalysis.com/article/performance-indicators-in-football.
[22] Ambrose, S. C., & Schnitzlein, N. (2017). What makes for the best rivalries in individual sports and how can marketers capitalize on them?. *Sport Marketing Quarterly*, 26(4).
[23] Cobbs, J., Sparks, B. D., & Tyler, B. D. (2017). Comparing rivalry effects across professional sports: National Football League fans exhibit most animosity. *Sport Marketing Quarterly*, 26(4), 235-246.
[24] Bowen, F. (2020, March 25). Perspective | Like Tom Brady, lots of great athletes have changed teams as they near retirement. The Washington Post. https://www.washingtonpost.com/lifestyle/kidspost/lots-of-great-athletes-like-tom-brady-have-changed-teams-as-they-near-retirement/2020/03/25/86a5d9ae-6469-11ea-845d-e35b0234b136_story.html.
[25] Hughes, M. D., Caudrelier, T., James, N., Redwood-Brown, A., Donnelly, I., Kirkbride, A., & Duschesne, C. (2012). Moneyball and soccer-an analysis of the key performance indicators of elite male soccer players by position.
[26] Hughes, M. D., & Bartlett, R. M. (2002). The use of performance indicators in performance analysis. *Journal of sports sciences*, 20(10), 739-754.
[27] Banker, S. (2020, September 3). No Industry Has A Better Key Performance Indicator Than Baseball. Forbes. https://www.forbes.com/sites/stevebanker/2020/09/02/no-industry-has-a-better-measure-of-what-it-takes-to-succeed-than-baseball/?sh=1a5a3b88240f.
[28] Onağ, Z., & Tepeci, M. (2014). Team effectiveness in sport teams: The effects of team cohesion, intra team communication and team norms on team member satisfaction and intent to remain. *Procedia-Social and Behavioral Sciences*, 150, 420-428.

8. Appendix

8.1. Network of Athletes for 200-metre men's butterfly (Task 2a)



The weighted directed network of relative ability for the 200-metre men’s butterfly, with top 3 athletes Michael Phelps, Melvin Stewart, Kristóf Milák being coloured red, orange, green respectively.

8.2. Data files used for Task 1 (tennis) and Task 2 (swimming)

8.2.1. Reformatted data for Task 1 (tennis)

Below is our reformatted data of the matches in the Grand Slam tournaments in 2018:

P1	P2	P1-1	P2-1	P1-2	P2-2	P1-3	P2-3	P1	P2	P1-1	P2-1	P1-2	P2-2	P1-3	P2-3
K Kanepi	S Williams	0	6	6	4	3	6	A Van Uytvanck	D Kasatkina	7-8	6-6	3	6	2	6
A Barty	Ka Pliskova	4	6	4	6			A Kerber	B Bencic	6	3	7-7	6-5		
S Stephens	E Mertens	6	3	6	3			Ka Pliskova	K Bertens	3	6	6-1	7-7		
A Sevastova	E Svitolina	6	3	1	6	6	0	J Gorges	D Vekic	6	3	6	2		
C Suarez Navarro	M Sharapova	6	4	6	3			S Williams	E Rodina	6	2	6	2		
M Keys	D Cibulkova	6	1	6	3			C Giorgi	E Makarova	6	3	6	4		
A Sabalenka	N Osaka	3	6	6	2	4	6	D Cibulkova	J Ostapenko	5	7	4	6		
M Vondrousova	L Tsurenko	7-7	6-3	5	7	2	6	D Kasatkina	A Kerber	3	6	5	7		
S Williams	Ka Pliskova	6	4	6	3			K Bertens	J Gorges	6	3	5	7	1	6
S Stephens	A Sevastova	2	6	3	6			S Williams	C Giorgi	3	6	6	3	6	4
C Suarez Navarro	M Keys	4	6	3	6			J Ostapenko	A Kerber	3	6	3	6		
N Osaka	L Tsurenko	6	1	6	1			J Gorges	S Williams	2	6	4	6		
S Williams	A Sevastova	6	3	6	0			A Kerber	S Williams	6	3	6	3		
M Keys	N Osaka	2	6	4	6			S Halep	E Mertens	6	2	6	1		
S Williams	N Osaka	2	6	4	6			A Kerber	C Garcia	6	2	6	3		
S Halep	N Osaka	6	3	6	2			G Muguruza	L Tsurenko	2	0				
B Strycova	Ka Pliskova	7-7	6-5	3	6	2	6	B Strycova	Y Putintseva	4	6	3	6		
S-w Hsieh	A Kerber	6	4	5	7	2	6	M Keys	M Buzarnescu	6	1	6	4		
M Keys	C Garcia	6	3	6	2			A Kontaveit	S Stephens	2	6	0	6		
P Martic	E Mertens	6-5	7-7	5	7			D Kasatkina	C Wozniacki	7-7	6-5	6	3		
D Allertova	E Svitolina	3	6	0	6			S Halep	A Kerber	6-2	7-7	6	3	6	2
A Kontaveit	C Suarez Navarro	6	4	4	6	6	8	G Muguruza	M Sharapova	6	2	6	1		
M Rybarikova	C Wozniacki	3	6	0	6			Y Putintseva	M Keys	6-5	7-7	4	6		
S Halep	Ka Pliskova	6	3	6	2			S Stephens	D Kasatkina	6	3	6	1		
A Kerber	M Keys	6	1	6	2			S Halep	G Muguruza	6	1	6	4		
E Mertens	E Svitolina	6	4	6	0			M Keys	S Stephens	4	6	4	6		
C Suarez Navarro	C Wozniacki	0	6	7-7	6-3	2	6	S Halep	S Stephens	3	6	6	4	6	1
S Halep	A Kerber	6	3	4	6	9	7								
E Mertens	C Wozniacki	3	6	6-2	7-7										
S Halep	C Wozniacki	6-2	7-7	6	3	4	6								
S-w Hsieh	D Cibulkova	4	6	1	6										

8.2.2. Sample of competition records for swimming

Below is an extract from our cleansed competition records of men's 200-metre butterfly:

Rank	Athlete	Time	Type	Year	Competition
1	MichaelPHELPS	1:53.36	Final	2016	Olympics
2	MasatoSAKAI	1:53.40	Final	2016	Olympics
3	TamasKENDERESI	1:53.62	Final	2016	Olympics
4	ChadLE CLOS	1:54.06	Final	2016	Olympics
5	DaiyaSETO	1:54.82	Final	2016	Olympics
6	ViktorBROMER	1:55.64	Final	2016	Olympics
7	LaszloCSEH	1:56.24	Final	2016	Olympics
8	LouisCROENEN	1:57.04	Final	2016	Olympics
1	LaszloCSEH	1:55.18	Semi	2016	Olympics
2	DaiyaSETO	1:55.28	Semi	2016	Olympics
3	MasatoSAKAI	1:55.32	Semi	2016	Olympics
4	LouisCROENEN	1:56.03	Semi	2016	Olympics
5	GrantIRVINE	1:56.07	Semi	2016	Olympics
6	ZhengWen QUAH	1:56.11	Semi	2016	Olympics
7	KaioALMEIDA	1:57.45	Semi	2016	Olympics
8	ZhuhaoLI	1:57.62	Semi	2016	Olympics
1	TamasKENDERESI	1:53.96	Semi	2016	Olympics
2	MichaelPHELPS	1:54.12	Semi	2016	Olympics
3	ChadLE CLOS	1:55.19	Semi	2016	Olympics
4	ViktorBROMER	1:55.59	Semi	2016	Olympics
5	EvgenyKOPTELOV	1:56.46	Semi	2016	Olympics
6	SimonSJODIN	1:56.71	Semi	2016	Olympics
7	LeonardoDE DEUS	1:56.77	Semi	2016	Olympics
8	JonathanGOMEZ	1:57.47	Semi	2016	Olympics
1	MichaelPHELPS	1:52.03	Final	2008	Olympics
2	LaszloCSEH	1:52.70	Final	2008	Olympics
3	TakeshiMATSUDA	1:52.97	Final	2008	Olympics
4	MossBURMESTER	1:54.35	Final	2008	Olympics

8.3. Python source code

8.3.1. Python source code used for finding the greatest female tennis player in 2018

```
import pandas as pd
import numpy as np
import networkx as nx
import matplotlib.pyplot as plt

df = pd.read_csv('data_files/tennis_data.csv')
df.head()

def cleanseNum(a, b):
  if type(a) == float:
    a = str(int(a))
  if type(b) == float:
    b = str(int(b))
  x = 0
  y = 0
  if a.isnumeric():
    x = int(a)
  else:
    x = int(a.split('-')[0])

  if b.isnumeric():
    y = int(b)
  else:
    y = int(b.split('-')[0])
  return x, y

def cmp(a, b):
```

```
  x, y = cleanseNum(a, b)

  if x > y:
    return 1
  else:
    return 0


def margin(a, b):
  x, y = cleanseNum(a, b)
  return x - y

def isNaN(x):
  return x != x

def get_unweighted_graph():
  H = nx.DiGraph()

  for i,row in df.iterrows():
    p1 = 0
    p2 = 0
    if cmp(row['P1-1'], row['P2-1']):
      p1 += 1
    else:
      p2 += 1
    if not isNaN(row['P1-2']):
      if cmp(row['P1-2'], row['P2-2']):
        p1 += 1
      else:
        p2 += 1
    if not isNaN(row['P1-3']):
      if cmp(row['P1-3'], row['P2-3']):
        p1 += 1
      else:
        p2 += 1
    if p1 > p2:
      H.add_edge(row['P1'], row['P2'])
    else:
      H.add_edge(row['P1'], row['P2'])

def get_weighted_graph():
  H = nx.DiGraph()

  for i,row in df.iterrows():
    w = 0
    w +=  margin(row['P1-1'], row['P2-1'])
    if not isNaN(row['P1-2']):
      w += margin(row['P1-2'], row['P2-2'])
    if not isNaN(row['P1-3']):
      w += margin(row['P1-3'], row['P2-3'])
    if H.has_edge(row['P1'], row['P2']):
      H[row['P1']][row['P2']]['weight'] += w
    else:
      H.add_edge(row['P1'], row['P2'], weight = w)
  H1 = H.copy()
  for u,v in H1.edges():
    if H1.has_edge(v,u):
      w = H1[u][v]['weight'] - H1[v][u]['weight']
      if w <= 0:
        H.remove_edge(u,v)
```

```
      else:
        H[u][v]['weight'] = w
    else:
      w = H1[u][v]['weight']
      if w == 0:
        H.remove_edge(u,v)
      elif w < 0:
        H.remove_edge(u,v)
        H.add_edge(v, u, weight = -1*w)
  return H

def weighted_mean(summ, cnt, tournaments):
  #return summ/len(tournaments)
  return summ/(cnt**1)

tournament_cnt={}#player name to list of tournaments played
def get_graph_v3():#the one with the averaging...
  global tournament_cnt

  H = nx.DiGraph()

  edges={}#map (p1, p2) to [total w, no. of matches...]

  for i,row in df.iterrows():
    w = 0
    cnt=1
    w +=  margin(row['P1-1'], row['P2-1'])
    if not isNaN(row['P1-2']):
      w += margin(row['P1-2'], row['P2-2'])
      cnt+=1
    if not isNaN(row['P1-3']):
      w += margin(row['P1-3'], row['P2-3'])
      cnt+=1
    p1,p2=row['P1'],row['P2']

    tournament=row['Game'][0]
    if (p1,p2) in edges:
      edges[(p1,p2)][0]+=w
      edges[(p1,p2)][1]+=cnt
      if tournament not in edges[(p1,p2)][2]:
        edges[(p1,p2)][2]+=tournament
    elif (p2,p1) in edges:
      edges[(p2,p1)][0]-=w
      edges[(p2,p1)][1]+=cnt
      if tournament not in edges[(p2,p1)][2]:
        edges[(p2,p1)][2]+=tournament
    else:
      edges[(p1,p2)]=[w,cnt, tournament]

    if p1 in tournament_cnt:
      if tournament not in tournament_cnt[p1]:
        tournament_cnt[p1]+=tournament
    else:
      tournament_cnt[p1]=tournament
    if p2 in tournament_cnt:
      if tournament not in tournament_cnt[p2]:
        tournament_cnt[p2]+=tournament
    else:
      tournament_cnt[p2]=tournament
    # if H.has_edge(row['P1'], row['P2']):
```

```
    #   H[row['P1']][row['P2']]['weight'] += w
    # else:
    #   H.add_edge(row['P1'], row['P2'], weight = w)

  for key in edges:
    p1,p2=key
    w,cnt,tournament=edges[key]
    if w<0:
      w= -w
      p1,p2 = p2,p1#swap the 2 players...
    # if w==0:
    #   continue
    weight = weighted_mean(w,cnt,tournament)
    H.add_edge(p1, p2, weight=weight)
  return H

def Floyd(graph,weighted=False):
  node_list=[]
  name_to_label={}
  for i,node in enumerate(graph.nodes()):
    node_list.append(node)
    name_to_label[node]=i
  adjmat=[[-1 for node in node_list] for node in node_list]
  for i in range(len(node_list)):
    adjmat[i][i]=0#create "self-loops"
  for u,v in graph.edges():
    p1=name_to_label[u]
    p2=name_to_label[v]
    adjmat[p1][p2]=graph[u][v]['weight']#adjmat expresses margin of win...

  for k in range(len(node_list)):
    for i in range(len(node_list)):
      for j in range(len(node_list)):
        if adjmat[i][k]==-1 or adjmat[k][j]==-1:
          continue
        if adjmat[i][j]==-1 or adjmat[i][j]>adjmat[i][k]+adjmat[k][j]:
          adjmat[i][j]=adjmat[i][k]+adjmat[k][j]

  new_graph = nx.DiGraph()
  for i in range(len(node_list)):
    for j in range(i+1,len(node_list)):
      if adjmat[i][j]==-1 and adjmat[j][i]==-1:
        continue
      if adjmat[i][j]==adjmat[j][i]:
        continue

      if adjmat[j][i]==-1 or adjmat[i][j]>adjmat[j][i]:
        if weighted:
              new_graph.add_edge(node_list[i],node_list[j],weight=adjmat[i][j]-
max(adjmat[j][i],0) )
        else:
          new_graph.add_edge(node_list[i],node_list[j],weight=1)
      else:
        if weighted:
              new_graph.add_edge(node_list[j],node_list[i],weight=adjmat[j][i]-
max(adjmat[i][j],0) )
        else:
          new_graph.add_edge(node_list[j],node_list[i],weight=1)
  return new_graph
```

```
#G = get_weighted_graph()
G = get_graph_v3()
#nx.draw(G,      width      =      0.3*np.array(list(nx.get_edge_attributes(G,
'weight').values())))

g2=Floyd(G,weighted=True)

degree_ratio = [((g2.out_degree(node, weight = 'weight')+1)/(g2.in_degree(node,
weight = 'weight')+1),node) for node in g2.nodes()]
degree_ratio.sort(reverse=True)
#degree_ratio
ranklist=[]
for i,(score, name) in enumerate(degree_ratio):
  print(f"{i+1:>3}. {round(score,2):<5}  {name}")
  ranklist.append((score,name))

xx=[]
yy=[]
names=[]

for i,(score,name) in enumerate(ranklist,1):
  xx.append(i)
  yy.append(score)
  names.append(name)

cutoff=10

plt.bar(xx[:cutoff],yy[:cutoff])
plt.xticks(xx[:cutoff],names[:cutoff],rotation='vertical')
plt.title("Degree ratio (d) for top 10 tennis athletes")
plt.ylabel("Degree ratio (d)")
#plt.show()
plt.savefig('Tennis graph.png',dpi=300,bbox_inches='tight')
```

8.3.2. Python source code used for finding the G.O.A.T. of men's 200m butterfly

```
a_i_range=[1,1.25]
b_i_range=[1,2]
c_i_range=[1,1.5]

"""##Import lib + read data from file"""

import pandas as pd
import numpy as np
import networkx as nx
import matplotlib.pyplot as plt

df = pd.read_csv('data_files/swimdata.csv')
df.head()

"""##Ability Calculation"""

ability_score={}

"""###Cleanse time method"""

def cleanse(s):
  while not s[0].isnumeric():
    s = s[1:]
```

```
  while not s[-1].isnumeric():
    s = s[:-1]
  return s

def gettime(s):
  s = cleanse(s)
  l = s.split(':')
  return float(l[0])*60+float(l[1])

"""###Ability calculation: the part with network"""

comp_records={}#(type,year,competition) : [(Name, Timing)]

for i,row in df.iterrows():
  comp=(row['Type'].strip(),row['Year'],row['Competition'].strip())
  if comp in comp_records:
    comp_records[comp].append( (row['Athlete'].strip(),gettime(row['Time'])) )
  else:
    comp_records[comp] = [ (row['Athlete'].strip(),gettime(row['Time'])) ]

edges={}#(name1, name2): [sum of differences, race_count]

for comp in comp_records:
  #comp = (type,year, competition)
  if comp[0]=='Semi':
    continue
  timings=comp_records[comp]#list of (name,timing)

  timings.sort(key=lambda x:x[1])
  timings=timings[:8]

  for i in range(0,len(timings)):#find every possible pair of athletes, i faster
than j
    for j in range(i+1,len(timings)):
      p1=timings[i][0]
      p2=timings[j][0]
      diff=abs(timings[i][1]-timings[j][1])
      if (p1,p2) in edges:#p1 is winning
        edges[(p1,p2)][0]+=diff
        edges[(p1,p2)][1]+=1
      elif (p2,p1) in edges:
        edges[(p2,p1)][0]-=diff
        edges[(p2,p1)][1]+=1
      else:
        edges[(p1,p2)]=[diff,1]

graph=nx.DiGraph()

def weighted_mean(weight,count):
  return weight/count

for key in edges:
    p1,p2=key
    w,cnt=edges[key]
    if w<0:
      w= -w
      p1,p2 = p2,p1#swap the 2 players...
    # if w==0:
    #   continue
    weight = weighted_mean(w,cnt)
```

```
    graph.add_edge(p1, p2, weight=weight)

G=graph

degree_ratio = [((G.out_degree(node, weight = 'weight')+1)/((G.in_degree(node,
weight = 'weight')+1)),node) for node in G.nodes()]
degree_ratio.sort(reverse=True)
# degree_ratio
for i,(score, name) in enumerate(degree_ratio):
   ability_score[name]=score
#   print(f"{i+1:>3}. {round(score,2):<5}  {name}")

"""##Other parts
1. Normalised weighted medal count
2. No. of times each person enters finals
3. World Record count
"""

player_data={}#name: weighted medal count b_i, WR_count sqrt(r)+1, competition
count c_i

"""###Normalised weighted medal cnt + WR_count"""

for i,row in df.iterrows():
  name=row['Athlete']
  if name not in player_data:
    player_data[name]=[0,0]
  rankk=int(row['Rank'])

  if row['Type']=='Final' and rankk<=3:
    player_data[name][0]+= (4-rankk)#weighted medal count
  if row['Time'].find('WR')!=-1:
    player_data[name][1]+=1

##Normalisation of medal count (b_i), changing r to sqrt(r)+1
minn=1e9
maxx=-1e9
for player in player_data:#find range of weighted medal count for
normalisation...
  player_data[player][0]=player_data[player][0]**0.5
  player_data[player][1]=1+(player_data[player][1]**0.5)
  minn=min(minn,player_data[player][0])
  maxx=max(maxx,player_data[player][0])

for player in player_data:
  unweighted=player_data[player][0]
  weighted=(b_i_range[1]-b_i_range[0])*( (unweighted-minn)/(maxx-minn) )
  player_data[player][0]=weighted+b_i_range[0]

##Computation of competition count (only counts)
comps = {row.Athlete: set() for i,row in df.iterrows()}
for i,row in df.iterrows():
  if row.Type == 'Final':
    comps[row.Athlete].add(row.Competition + ' ' + str(row.Year))

##Normalisation of competition count (c_i), into 1-1.5
minn=1e9
maxx=-1e9
for athlete in comps:#find range of weighted medal count for normalisation...
  minn=min(minn,len(comps[athlete]))
```

```
  maxx=max(maxx,len(comps[athlete]))

for athlete in comps:
  unweighted=len(comps[athlete])
  weighted= (unweighted-minn)/(maxx-minn)
  comps[athlete]=weighted*(c_i_range[1]-c_i_range[0])+c_i_range[0]
  player_data[athlete].append(comps[athlete])

"""##Rivalry"""

rivals={} #(name1, name2): [total,cnt]
rival_score={}#name: count
for athlete in player_data:
  rival_score[athlete]=(0,0)

cur_comp=(None,None,None)#type, year, competition
medallists=[]
prev_time=None

for i,row in df.iterrows():
  if row['Type'].strip().lower()[:5]!='final':
    continue
  comp=(row['Type'].strip(),row['Year'],row['Competition'].strip())
  if comp!=cur_comp:
    medallists=[]
    cur_comp=comp

  if row['Rank']<=3:
    cur_time=gettime(row['Time'])
    for prev_athlete,prev_time in medallists:
      pair=
( min(prev_athlete,row['Athlete']),max(prev_athlete,row['Athlete']) )
      if pair in rivals:
        rivals[pair][0]+=abs(cur_time-prev_time)
        rivals[pair][1]+=1
      else: rivals[pair]=[abs(cur_time-prev_time),1]
    medallists.append( (row['Athlete'],cur_time) )

minn=0
maxx=0#to help with normalisation...

for pair in rivals:
  avg_diff=round(rivals[pair][0]/rivals[pair][1],3)
  score=max(1-(2*avg_diff),0)
  maxx=max(score,maxx)
  for athlete in pair:
    if rival_score[athlete][0]<score:
      rival_score[athlete]=(score,pair)

#Normalisation
for athlete in rival_score:
  rival_score[athlete]=a_i_range[0]+((a_i_range[1]-a_i_range[0])
*(rival_score[athlete][0]-minn)/(maxx-minn))

"""##Putting all data together..."""

goatness={}

for athlete in ability_score:
  d_i=ability_score[athlete]
```

```
  b_i,r,c_i=player_data[athlete]
  a_i=rival_score[athlete]

  goatness[athlete]=d_i*b_i*r*c_i*a_i

"""###Generate ranklist..."""

ranklist=[]
for athlete in goatness:
  ranklist.append( (goatness[athlete],athlete) )
ranklist.sort(reverse=True)

for i,(score,name) in enumerate(ranklist,1):
  print(f"{(i):>3}.  {name:<30}   {round(score,2):<7}  {rival_score[name]}")

"""###Plot graph based on ranklist"""

xx=[]
yy=[]
names=[]

for i,(score,name) in enumerate(ranklist,1):
  xx.append(i)
  yy.append(score)
  names.append(name)

cutoff=10

colors=['blue']+['blue']*(cutoff-1)#in case change colour

plt.bar(xx[:cutoff],yy[:cutoff])#,color=colors)
plt.xticks(xx[:cutoff],names[:cutoff],rotation='vertical')
plt.title("G.O.A.T.-ness score for top 10 swimming athletes")
plt.ylabel("G.O.A.T.-ness score")
#plt.show()

plt.savefig('Swimming graph.png',dpi=300,bbox_inches='tight')
```

8.3.3. Python source code used for sensitivity analysis for swimming

```
shot_cnt=1000
rank_cnts=[{},{},{}]

for x in range(shot_cnt):
    if (x%100)==0:
        print(x)

    a_i_range=[1,1.25]
    b_i_range=[1,2]
    c_i_range=[1,1.5]

    """##Import lib + read data from file"""

    import pandas as pd
    import numpy as np
    import networkx as nx
    #import matplotlib.pyplot as plt

    df = pd.read_csv('data_files/swimdata.csv')
```

```
df.head()

import random
lis=[]
for i,row in df.iterrows():
  lis.append(i)
random.shuffle(lis)
lis=lis[:len(lis)//10]

"""##Ability Calculation"""

ability_score={}

"""###Cleanse time method"""

def cleanse(s):
  while not s[0].isnumeric():
    s = s[1:]
  while not s[-1].isnumeric():
    s = s[:-1]
  return s

def gettime(s):
  s = cleanse(s)
  l = s.split(':')
  return float(l[0])*60+float(l[1])

"""###Ability calculation: the part with network"""

comp_records={}#(type,year,competition) : [(Name, Timing)]

for i,row in df.iterrows():
  if i in lis:
    continue
  comp=(row['Type'].strip(),row['Year'],row['Competition'].strip())
  if comp in comp_records:
    comp_records[comp].append( (row['Athlete'].strip(),gettime(row['Time']))
)
  else:
    comp_records[comp] = [ (row['Athlete'].strip(),gettime(row['Time'])) ]

edges={}#(name1, name2): [sum of differences, race_count]

for comp in comp_records:
  #comp = (type,year, competition)
  if comp[0]=='Semi':
    continue
  timings=comp_records[comp]#list of (name,timing)

  timings.sort(key=lambda x:x[1])
  timings=timings[:8]

  for i in range(0,len(timings)):#find every possible pair of athletes, i
faster than j
    for j in range(i+1,len(timings)):
      p1=timings[i][0]
      p2=timings[j][0]
      diff=abs(timings[i][1]-timings[j][1])
      if (p1,p2) in edges:#p1 is winning
        edges[(p1,p2)][0]+=diff
```

```
          edges[(p1,p2)][1]+=1
        elif (p2,p1) in edges:
          edges[(p2,p1)][0]-=diff
          edges[(p2,p1)][1]+=1
        else:
          edges[(p1,p2)]=[diff,1]

    graph=nx.DiGraph()

    def weighted_mean(weight,count):
      return weight/count

    for key in edges:
        p1,p2=key
        w,cnt=edges[key]
        if w<0:
          w= -w
          p1,p2 = p2,p1#swap the 2 players...
        # if w==0:
        #   continue
        weight = weighted_mean(w,cnt)
        graph.add_edge(p1, p2, weight=weight)

    G=graph
    #nx.draw(G, width = 0.3*np.array(list(nx.get_edge_attributes(G,
'weight').values())))

    #nx.write_graphml(G, f"/content/drive/MyDrive/IMMC 2021/graph_files/
swimming3.graphml")

    degree_ratio = [((G.out_degree(node, weight = 'weight')+1)/
((G.in_degree(node, weight = 'weight')+1)),node) for node in G.nodes()]
    degree_ratio.sort(reverse=True)
    # degree_ratio
    for i,(score, name) in enumerate(degree_ratio):
       ability_score[name]=score
    #   print(f"{i+1:>3}. {round(score,2):<5}  {name}")

    """##Other parts
    1. Normalised weighted medal count
    2. No. of times each person enters finals
    3. World Record count
    """

    player_data={}#name: weighted medal count b_i, WR_count sqrt(r)+1,
competition count c_i

    """###Normalised weighted medal cnt + WR_count"""

    for i,row in df.iterrows():
      name=row['Athlete']
      if name not in player_data:
        player_data[name]=[0,0]
      rankk=int(row['Rank'])

      if i in lis:
        continue

      if row['Type']=='Final' and rankk<=3:
        player_data[name][0]+= (4-rankk)#weighted medal count
```

```
      if row['Time'].find('WR')!=-1:
        player_data[name][1]+=1

    ##Normalisation of medal count (b_i), changing r to sqrt(r)+1
    minn=1e9
    maxx=-1e9
    for player in player_data:#find range of weighted medal count for
normalisation...
      player_data[player][0]=player_data[player][0]**0.5
      player_data[player][1]=1+(player_data[player][1]**0.5)
      minn=min(minn,player_data[player][0])
      maxx=max(maxx,player_data[player][0])

    for player in player_data:
      unweighted=player_data[player][0]
      weighted=(b_i_range[1]-b_i_range[0])*( (unweighted-minn)/(maxx-minn) )
      player_data[player][0]=weighted+b_i_range[0]

    ##Computation of competition count (only counts)
    comps = {row.Athlete: set() for i,row in df.iterrows()}
    for i,row in df.iterrows():
      if i in lis:
        continue
      if row.Type == 'Final':
        comps[row.Athlete].add(row.Competition + ' ' + str(row.Year))

    ##Normalisation of competition count (c_i), into 1-1.5
    minn=1e9
    maxx=-1e9
    for athlete in comps:#find range of weighted medal count for
normalisation...
      minn=min(minn,len(comps[athlete]))
      maxx=max(maxx,len(comps[athlete]))

    for athlete in comps:
      unweighted=len(comps[athlete])
      weighted= (unweighted-minn)/(maxx-minn)
      comps[athlete]=weighted*(c_i_range[1]-c_i_range[0])+c_i_range[0]
      player_data[athlete].append(comps[athlete])

    """##Rivalry"""

    rivals={} #(name1, name2): [total,cnt]
    rival_score={}#name: count
    for athlete in player_data:
      rival_score[athlete]=(0,0)

    cur_comp=(None,None,None)#type, year, competition
    medallists=[]
    prev_time=None

    for i,row in df.iterrows():
      if i in lis:
        continue
      if row['Type'].strip().lower()[:5]!='final':
        continue
      comp=(row['Type'].strip(),row['Year'],row['Competition'].strip())
      if comp!=cur_comp:
        medallists=[]
        cur_comp=comp
```

```
      if row['Rank']<=3:
        cur_time=gettime(row['Time'])
        for prev_athlete,prev_time in medallists:
          pair=
( min(prev_athlete,row['Athlete']),max(prev_athlete,row['Athlete']) )
          if pair in rivals:
            rivals[pair][0]+=abs(cur_time-prev_time)
            rivals[pair][1]+=1
          else: rivals[pair]=[abs(cur_time-prev_time),1]
        medallists.append( (row['Athlete'],cur_time) )

    minn=0
    maxx=0#to help with normalisation...

    for pair in rivals:
      avg_diff=round(rivals[pair][0]/rivals[pair][1],3)
      score=max(1-(2*avg_diff),0)
      maxx=max(score,maxx)
      for athlete in pair:
        if rival_score[athlete][0]<score:
          rival_score[athlete]=(score,pair)

    #Normalisation
    for athlete in rival_score:
      rival_score[athlete]=a_i_range[0]+((a_i_range[1]-a_i_range[0])
*(rival_score[athlete][0]-minn)/(maxx-minn))

    """##Putting all data together..."""

    goatness={}

    for athlete in ability_score:
      d_i=ability_score[athlete]
      b_i,r,c_i=player_data[athlete]
      a_i=rival_score[athlete]

      goatness[athlete]=d_i*b_i*r*c_i*a_i

    """###Generate ranklist..."""

    ranklist=[]
    for athlete in goatness:
      ranklist.append( (goatness[athlete],athlete) )
    ranklist.sort(reverse=True)

    for i,(score,name) in enumerate(ranklist[:3]):
        if name in rank_cnts[i]:
            rank_cnts[i][name]+=1
        else:
            rank_cnts[i][name]=1

for mp in rank_cnts:
    print(mp)
```