5-2022

Implementing the CMS+ Sports Rankings Algorithm in a JavaFx Environment

Luke Welch
University of Arkansas, Fayetteville

Implementing the CMS+ Sports Rankings Algorithm in a JavaFx Environment

A thesis submitted in partial fulfillment of
the requirements for the degree of Bachelor
of Science in Industrial Engineering with
Honors

By
Luke Welch

Spring 2022

University of Arkansas
Department of Industrial Engineering

Thesis Advisor: C. Richard Cassady, Ph.D.

Thesis Reader: Chase E. Rainwater, Ph.D.

Abstract

Every year, sports teams and athletes get cut from championship opportunities because of their rank. While this reality is easier to swallow if a team or athlete is distant from the cut, it is much harder when they are right on the edge. Many times, it leaves fans and athletes wondering, “Why wasn’t I ranked higher? What factors when into the ranking? Are the rankings based on opinion alone?” These are fair questions that deserve an answer. Many times, sports rankings are derived from opinion polls. Other times, they are derived from a combination of opinion polls and measured performance. This leads to the unfortunate reality that, many times, athletes and the public don’t truly know what goes into the rankings behind the scenes.

A sports ranking system that quantifies human bias and reveals it in the process of ranking sports teams is a solution to this problem. Allowing athletes and the public to see exactly how much weight is given to every aspect of their performance creates a system that alleviates the hiddenness and secretive nature of many ranking systems. The CMS+ algorithm is a college football ranking approach that allows users to specify the importance of certain factors and generate rankings based on those factors. This research aims to develop a Java implementation of the already defined CMS+ algorithm and redefine the approach for Mixed Martial Arts.

Introduction

The Ultimate Fighting Championship (UFC) is a division of Mixed Martial Arts (MMA) that consists of the most talented fighters in the world. It offers the most money for fighters, the biggest promotions, and the largest venues of any other MMA division. Finding success in amateur divisions is the most common route to securing a fight contract in the UFC [1]. Once a contract is secured, however, the competition is only fiercer. While the ultimate goal for UFC fighters is to have a shot at the championship belt, they are also trying to score the biggest fights possible on their way. With a large pool of fighters waiting for their chance to secure their next fight, it is important to select the fighters who are the most deserving.

Talent, record, and significant wins are the most important factors that set fighters apart from each other. In an attempt to quantify these factors, the UFC currently uses a ranking system that is comprised of a voting panel made up of media members who are asked to vote for who they feel are the top fighters in the UFC by weight-class and pound-for-pound. Then, the average ratings for each fighter are calculated, resulting in a rank for every fighter. This process happens behind closed doors, resulting in a private, biased, and subjective system [2].

Because the voting process is hidden from the public and the fighters, it is unclear what factors are going into the ranks. Recency bias, the tendency to give a fighter a higher rank due to a recent win, is an example of human bias that plays a role in these rankings. There is also a tendency to give a fighter a higher rank based on their likeliness to end the fight early. These human biases, mixed with the private nature of the voting process, lead to a ranking system that is unfair and unclear both to fighters and fans.

To create a fair ranking system for UFC fighters, it is important to identify biases that affect the system. Once biases are identified, an optimization model that implements user-specified factors will be able to apply those biases fairly. This will result in a ranking system that effectively identifies bias, implements user-specified factors, and optimizes UFC fighter ranks.

Literature Review

Outside of the realm of UFC, there are many different approaches that have been studied in an attempt to fairly rank sports teams. Neural networks, machine learning, regression analysis, and linear systems are just a few approaches that have been applied to ranking systems. While the CMS+ system is designed as a quadratic-assignment problem (QAP) [3], it is useful to be familiar with other notable methodologies for ranking sports teams.

The Massey method is a sports ranking system that was introduced by Kenneth Massey in 1997 [4]. This method is based on the idea that the difference between two teams' ranks can be used to predict the point differential in a head-to-head competition. These point differentials can then be aggregated over all games for each team and result in a unique solution. Additionally, instead of predicting (and using) point differentials for developing a unique solution, a team's win-loss record can be used to develop a unique solution of ranks.

The Colley method is a "bias free" sports ranking system that was introduced by W. N. Colley in 2002 [4]. This method uses a modified winning percentage for each team by calculating the number of games won based on how many games the team has played. Each team begins with a Colley rating that is updated based on the modified win percentages of the teams; a solution vector outputs the rankings of the teams based on a combination of the Colley rating and win percentage of each team.

The Markov method is a more universal approach that has many applications [4]. One application relies on a node-network where each team represents a node and each edge represents teams that have played each other. A random walk is defined on the nodes where the walker moves from team to team with each team having a win probability based on head-to-head wins and overall wins. Then, by defining an eigenvector and performing matrix calculations, a solution vector is developed that includes the ranks of each team.

An extension of the Markov methodology is Crowd-Ranking, where the weaker alternative casts a vote for the stronger alternative for each game [5]. The votes populate a dominance matrix that is used to rank the alternatives. The Crowd-Ranking methodology relies on the idea that the losing alternative places one vote for the winning alternative.

There are fewer ranking methodologies in the world of Mixed Martial Arts. One notable methodology developed by Gaurav Sharma and Atul Kumar Uttam relies on a multilayer neural network model for predicting the winner of a Mixed Martial Arts fight [6]. The neural network is composed of an input layer, two hidden layers, and an output layer. The input layer consists of 117 nodes which correspond to the entire number of features. The hidden layers consist of 50 and 10 nodes. After calculating a weighted sum plus bias when an incoming input is received, an activation function is employed and the predicted winner is outputted.

The CMS+ Algorithm

The CMS+ system is designed as a quadratic-assignment problem (QAP), which is a combinatorial optimization problem that has received significant attention in the operations research literature [7]. A QAP is proven to be an NP-hard problem [8], meaning that computational time required to solve the problem to optimality using any known algorithm grows exponentially as the number of assignments goes beyond a small number [9]. A QAP begins with *n* facilities and *n* locations, with a distance assigned to each pair of locations and weight assigned to each pair of facilities. The problem assigns all facilities to different locations with the goal of minimizing the weighted sum of the distances across all pairs of located facilities. An example quadratic assignment problem is presented in Figure 1.

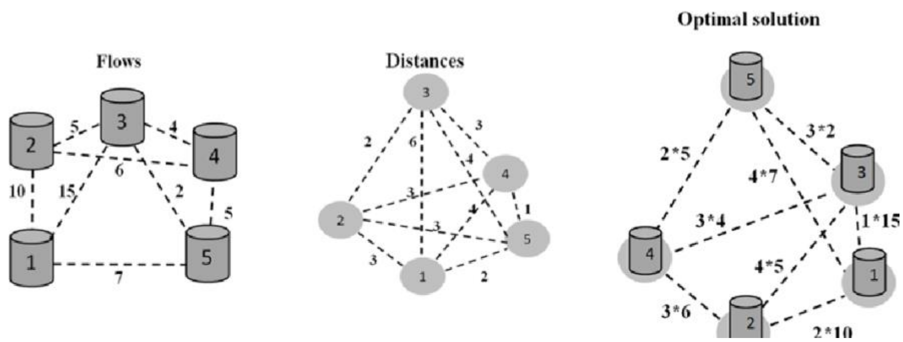


Figure 1. The Quadratic Assignment Problem Example [10]

In this example, the *Flows* diagram represents the weights assigned to each pair of facilities, with the dashed line numbers representing the weight, and the object numbers representing the facility. The *Distances* diagram represents the distance assigned to each pair of locations, with the dashed line numbers representing the distance, and the object numbers representing the location. The *Optimal Solution* diagram represents the assigned location of each facility such that the weighted sum of the distances across all pairs of the facilities is minimized.

The CMS+ system models the problem of ranking college football teams by replacing the facilities with teams and replacing locations with ranking positions. Since college football contains many teams, the computational complexity is very high for this QAP. A solution is to use a heuristic, an approach that can obtain acceptable results within a reasonable time frame but cannot guarantee an optimal solution [11]. Once the problem is set up, the user then specifies the importance of the degree of victory and relative distance between ranks. Then, by implementing nonuniform weighting to specific factors [12], factors like home-field advantage, overtime, margin of victory, and late-season games are implemented into the model. After fine tuning the parameters, the model outputs a set of rankings that are close to optimal.

Methodology

Java Implementation of CMS+ System

The CMS+ system is implemented in Excel's VBA software and can be executed through a macro-enabled button in the Excel workbook. Once executed, the output of the college football rankings and many statistics can be viewed on specific sheets within the workbook. The CMS+ system calculates degree of victory by allowing the user to specify the weight of several different factors related to the assignment of ranks: head-to-head victories, transitive victories, conference champions, and AP (Associated Press) Points are all user-specified, weighted factors that are considered in the ranking. Further, the user also specifies the relative distance parameter that determines how important the separation between certain ranks is.

In order to enhance the CMS+ application with faster performance, more features, and a friendlier user-interface, we decided to implement the application in Java. While there are many similarities between Excel VBA and Java language, implementing the CMS+ application in Java was not a simple "copy-and-paste" approach. There were a few important differences that between languages that had to be considered when implementing the application.

To begin the implementation process, I first created a JavaFx User-Interface (UI) that allows the user to view the games, teams, and conferences files (Figures 2, 3, and 4) that are being used to calculate rankings. Since the information in these files is simply represented in the sheets of the Excel application, a similar way to view the information in Java was a good first step. The UI includes three buttons that are linked to separate methods, each derived from a class containing information associated with each file. Each button takes the user to a new screen with a view of either the games, teams, or conferences file.

Teams

ID	Team	Conference	Division	CFP	APRank	APPoints
1	AirForce	7	1	26	24	97
2	Akron	6	1	26	36	0
3	Alabama	9	2	13	9	1011
4	Alabama-Birmingham	5	2	26	36	0
5	AppalachianState	10	1	20	20	355
6	Arizona	8	2	26	36	0
7	ArizonaState	8	2	26	31	5
8	Arkansas	9	2	26	36	0
9	ArkansasState	10	2	26	36	0
10	Army	11	1	26	36	0
11	Auburn	9	2	12	9	1011
12	BallState	6	2	26	36	0
13	Baylor	3	1	7	8	1039
14	BoiseState	7	1	19	18	500

Back

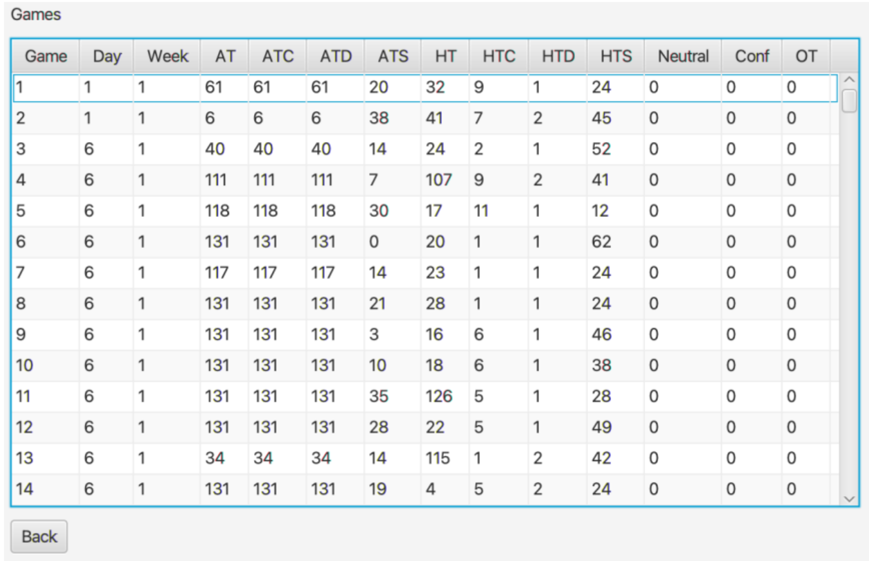
Figure 2. Teams table detailing information associated with each team.

Conferences

ID	Conference	Num_Teams	Num_Divisions	Division1	Division2	Champs	Champ1
1	AAC	12	2	East	West	1	60
2	ACC	14	2	Atlantic	Coastal	1	24
3	Big12	10	1	NA	NA	1	84
4	BigTen	14	2	East	West	1	83
5	ConferenceUSA	14	2	East	West	1	33
6	MAC	12	2	East	West	1	62
7	MWC	12	2	Mountain	West	1	14
8	Pac12	12	2	North	South	1	87
9	SEC	14	2	East	West	1	54
10	SunBelt	10	2	East	West	1	5
11	Independent	6	1	Na	NA	0	0

Back

Figure 3. Conferences table detailing information associated with each conference.


Games

Game	Day	Week	AT	ATC	ATD	ATS	HT	HTC	HTD	HTS	Neutral	Conf	OT
1	1	1	61	61	61	20	32	9	1	24	0	0	0
2	1	1	6	6	6	38	41	7	2	45	0	0	0
3	6	1	40	40	40	14	24	2	1	52	0	0	0
4	6	1	111	111	111	7	107	9	2	41	0	0	0
5	6	1	118	118	118	30	17	11	1	12	0	0	0
6	6	1	131	131	131	0	20	1	1	62	0	0	0
7	6	1	117	117	117	14	23	1	1	24	0	0	0
8	6	1	131	131	131	21	28	1	1	24	0	0	0
9	6	1	131	131	131	3	16	6	1	46	0	0	0
10	6	1	131	131	131	10	18	6	1	38	0	0	0
11	6	1	131	131	131	35	126	5	1	28	0	0	0
12	6	1	131	131	131	28	22	5	1	49	0	0	0
13	6	1	34	34	34	14	115	1	2	42	0	0	0
14	6	1	131	131	131	19	4	5	2	24	0	0	0

Back

Figure 4. Games table detailing information associated with each game.

After the files were read into the Java application and viewable to the user, I began implementing the methods that define degree of victory. First, I declared and initialized all the necessary variables and arrays. Then, I implemented four methods that define degree of victory: head-to-head victories, transitive victories, conference champions, and AP Points. The weights of these four parameters are defined by the user to allow a customizable ranking system that reveals the bias associated with the rankings. Next, I implemented the method to define the relative distance between ranks. For example, the measure for the difference between rank 1 and 2 vs. the difference between rank 129 and 130.

In the CMS+ application, once relative distance and degree of victory are defined and calculated, there is a feature that allows the user to evaluate the fitness of a certain ranking of teams. For example, if the user believes team 1 should be ranked 1, team 2 should be ranked 2, etc., they are able to calculate the fitness for those rankings based on the degree of victory and relative distance metrics calculated for each team. Therefore, once I implemented the degree of victory and relative distance methods in my Java application, I implemented the fitness method. I created three sample ranking combinations and compared the fitness for each ranking between Excel and Java. The results are shown in Figure 5.

Team ID	Rank	Team ID	Rank	Team ID	Rank
1	1	1	130	1	62
2	2	2	129	...	...
...	...	...	...	69	130
130	130	130	1	70	61
				...	...
				130	1
Fitness = 8071.26		Fitness = -8477.51		Fitness = -5190.85	

Figure 5. Three sample ranking combinations and their calculated fitness. The fitness values were the same for both the Excel VBA application and Java application.

After validating that the fitness values were the same for all three sample ranking combinations, ensuring that the relative distance and degree of victory were being calculated correctly, I began implementing the heuristic method that suggests a near optimal ranking based on the user-specified weights. The heuristic method proved to be the most challenging method to implement because of the randomness associated with the method. For example, the random solutions for the first iteration in the Excel VBA application were not necessarily the same random solutions for the first iteration in the Java application. Therefore, I performed further experimentation within the method by eliminating the randomness from the method in both Excel VBA and Java with the goal of comparing the methods "one-to-one." This experimentation is still underway. As such, though the Java application runs and produces an output of rankings, the results are incorrect and further work is required to correct the heuristic method in Java.

Redefining CMS+ approach for Mixed Martial Arts

Though degree of victory could prove more challenging to define in MMA, there are many similarities that would provide a relatively straightforward approach to redefining the CMS+ system for Mixed Martial Arts. Rather than game data for football games, fight match data could be used. Rajeev Warrier created a database that includes all UFC fight data from 1993 to 2021, with statistics of each fighter and each match during that time frame [13]. This would be a great starting place for observing fight data and would help accomplish the redefining of the system, as head-to-head fight outcomes would be easily accessible and the same CMS+ method could be used for fight data.

Another similarity between the college football CMS+ system and a Mixed Martial Arts system would be transitive victories. For example, the CMS+ system recognizes and considers when the following scenario occurs: team A beats team B; team B beats team C; team C beats team A. Though this scenario would most likely be less common in Mixed Martial Arts, it could be handled the same, nonetheless.

The CMS+ system implements AP Points for each team as a weighted factor when ranking teams. Though MMA does not have an AP Points system, the UFC currently uses a ranking system that is comprised of a voting panel made up of media members who are asked to vote for who they feel are the top fighters in the UFC by weight-class and pound-for-pound. These rankings could be used as a weighted factor for MMA in a similar fashion as college football team AP Points.

The last metric that the CMS+ system currently uses to define degree of victory is conference champs. This could be implemented in MMA by defining the “conference champ” as the reigning champ in each respective weight class. While the definition of degree of victory for

MMA could be relatively straightforward, there are some challenges that should be taken into consideration when redefining the system.

Due to the nature of the MMA (or UFC) sport, fighters fight very infrequently. Therefore, the data gets old much faster; a fighter could be involved in three fights in three years due to an injury or intense training. Therefore, I believe creating a metric like relative distance that measures the importance of recent fights vs. old fights would be helpful.

Another challenge is the fact that fights are not always fought within their weight class. For example, there have been cases where fighters miss their weight class (which should disqualify them from fighting) yet fight anyway. This begs the question: should a fighter who won while being heavier than their weight class be given the same amount of victory as if they met the weight?

In MMA, there are also different kinds of victories. Rather than a team either winning or losing, a fighter can win by knockout, submission, judges scorecards, etc. Should fighters be given more credit for winning by knockout than judges scorecards? These are considerations I believe are important when redefining the CMS+ system for MMA.

References

[1] "How To Get Into UFC? [And Become A UFC Fighter]", *Way of Martial Arts*, 2021. [Online]. Available: https://wayofmartialarts.com/is-it-hard-to-get-in-the-ufc-and-how-many-years-does-it-take-to-become-a-ufc-fighter/. [Accessed: 01- Oct- 2021].

[2] B. Dodge, "A Look Into The UFC Rankings System - Who is doing the voting?", *MyMMANews.com*, 2021. [Online]. Available: https://mymmanews.com/look-ufc-rankings-system/. [Accessed: 01- Oct- 2021].

[3] C. Cassady, L. Maillart and S. Salman, "Ranking Sports Teams: A Customizable Quadratic Assignment Approach", Interfaces, vol. 35, no. 6, pp. 497-510, 2005. Available: 10.1287/inte.1050.0171.

[4] S. Devlin and T. Treloar, "A network diffusion ranking family that includes the methods of Markov, Massey, and Colley", *Journal of Quantitative Analysis in Sports*, vol. 14, no. 3, pp. 91-101, 2018. Available: 10.1515/jqas-2017-0098.

[5] Vaziri, B., Dabadghao, S., Yih, Y., Morin, T. L., & Lehto, M. (2017). Crowd-ranking: A Markov-based method for ranking alternatives. Operational Research, 20(1), 279–295. https://doi.org/10.1007/s12351-017-0324-7

[6] Sharma, G., & Uttam, A. K. (2021). Multilayer neural network model for mixed martial arts winner prediction. *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*. https://doi.org/10.1109/iscon52037.2021.9702452

[7] E. Loiola, N. de Abreu, P. Boaventura-Netto, P. Hahn and T. Querido, "A survey for the quadratic assignment problem", *European Journal of Operational Research*, vol. 176, no. 2, pp. 657-690, 2007. Available: 10.1016/j.ejor.2005.09.032.

[8] S. Sahni and T. Gonzalez, "P-Complete Approximation Problems", *Journal of the ACM*, vol. 23, no. 3, pp. 555-565, 1976. Available: 10.1145/321958.321975.

[9] M. Bashiri and H. Karimi, "Effective heuristics and meta-heuristics for the quadratic assignment problem with tuned parameters and analytical comparisons", *Journal of Industrial Engineering International*, vol. 8, no. 1, 2012. Available: 10.1186/2251-712x-8-6.

[10] L. Shawky, M. Metwally and A. Zaied, "Quadratic Assignment Problem: A survey and Applications", *Semantic Scholar*, 2015. [Online]. Available: https://www.semanticscholar.org/paper/Quadratic-Assignment-Problem%3A-A-survey-and-Shawky-Metwally/247b45613d6d3b6961fdad44f9e7fefb70fd3e82. [Accessed: 01-Oct- 2021].

[11] T. Achary, S. Pillay, S. Pillai, M. Mqadi, E. Genders and A. Ezugwu, "A performance study of meta-heuristic approaches for quadratic assignment problem", *Concurrency and Computation: Practice and Experience*, vol. 33, no. 17, 2021. Available: 10.1002/cpe.6321.

[12] T. Chartier, E. Kreutzer, A. Langville and K. Pedings, "Sports Ranking with Nonuniform Weighting", *Journal of Quantitative Analysis in Sports*, vol. 7, no. 3, 2011. Available: 10.2202/1559-0410.1290.

[13] R. Warrier, "UFC-Fight historical data from 1993 to 2021", *Kaggle.com*, 2021. [Online]. Available: https://www.kaggle.com/rajeevw/ufcdata?select=raw_fighter_details.csv. [Accessed: 01- Oct- 2021].

Appendix

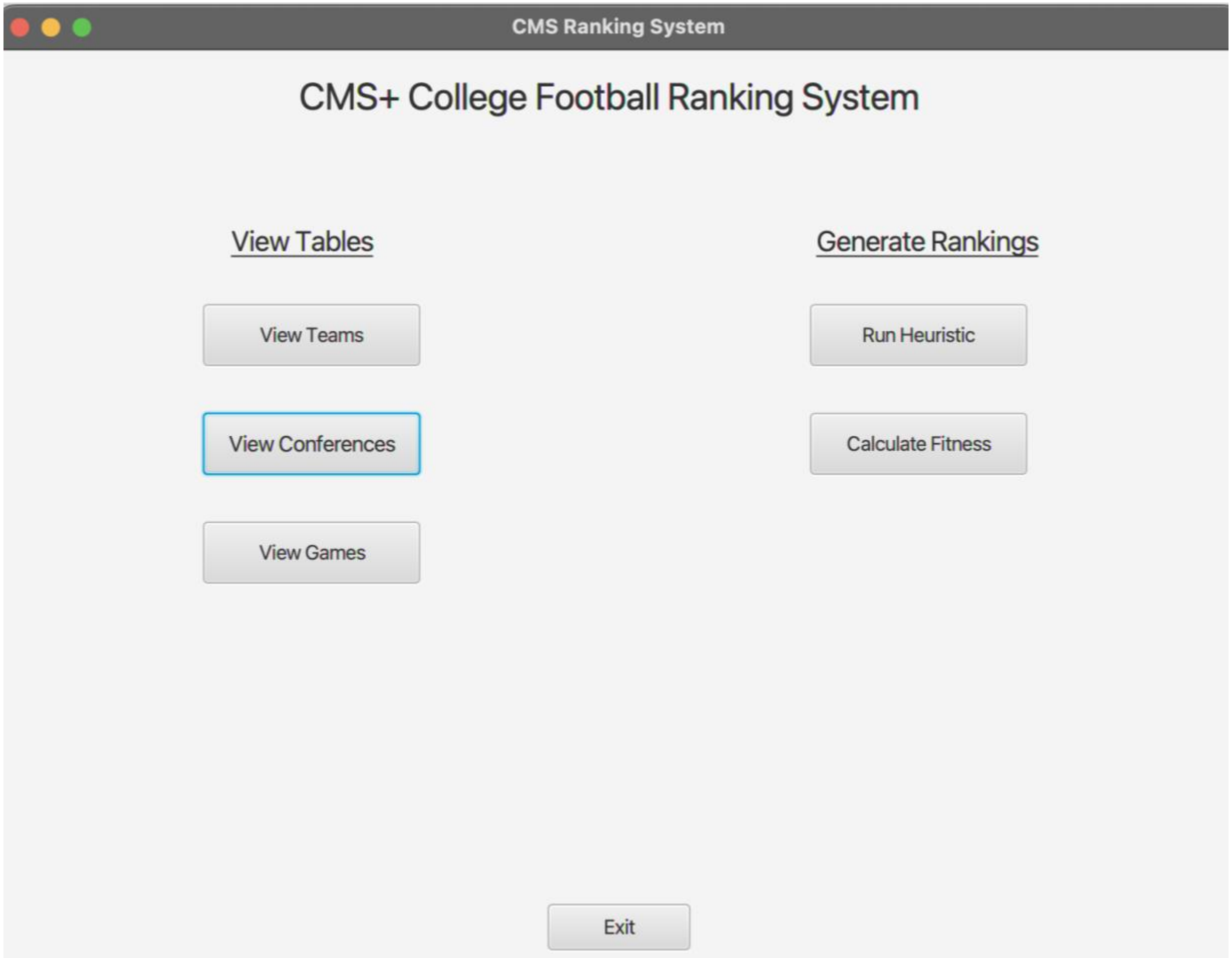


Figure 6. JavaFx Graphical User Interface (GUI) for CMS+ Application