A Network-Driven Methodology for Sports Ranking and Prediction

Vincent Xia[1], Kavirath Jain[1], Akshay Krishna[2], and Christopher G. Brinton[2]
[1]Department of Operations Research and Financial Engineering, Princeton University
[2]Department of Electrical Engineering, Princeton University

***Abstract*—Recent years have seen increasing interest in ranking elite athletes and teams in professional sports leagues, and in predicting the outcomes of games. In this work, we draw an analogy between this problem and one in the field of search engine optimization, namely, that of ranking webpages on the Internet. Motivated by the famous PageRank algorithm, our TeamRank methods define directed graphs of sports teams based on the observed outcomes of individual games, and use these networks to infer the importance of teams that determines their rankings. In evaluating these methods on data from recent seasons in the National Football League (NFL) and National Basketball Association (NBA), we find that they can predict the outcomes of games with up to 70% accuracy, and that they provide useful rankings of teams that cluster by league divisions. We also propose some extensions to TeamRank that consider overall team win records and shifts in momentum over time.**

I. Introduction

The abundance of data captured by professional sports leagues in the past several years has motivated a large body of research in data science and sports analytics. Sports analysts and writers often try to estimate metrics such as strength of victory (SOV) and strength of schedule (SOS) to rank teams and players in certain leagues. The resulting analytics have broad implications, from decisions on player trading to the placing of financial bets.

While several techniques – especially those in machine learning – have been well researched in the context of sports, the use of network modeling for sports leagues is relatively understudied. Consider Google's PageRank algorithm [1], which produces rankings of webpages by factoring in both their absolute and relative importance as determined from webgraphs. In ranking sports teams, we may similarly seek to factor in both absolute and relative measures of strength. Motivated by this, we apply variations of the PageRank algorithm to network models of sports leagues to generate team rankings and predictions.

A. Related Work

A plethora of research and innovation has been undertaken in general analytics for elite professional sports. Various novel metrics have been defined to quantify aspects of player performance, for example, in basketball [2] and in ice hockey [3]. These works have been primarily focused on the skills and strengths of individual players and their impact on team performance. Our work is different, however, because we study the problem of turning analytics to match predictions.

Machine learning has become a popular method for predicting sports outcomes. Several recent papers have described the applicability of machine learning techniques such as Bayesian models and deep neural networks to both game outcomes [4]–[6] as well as in-game strategies and plays [7], [8]. Despite being possibly more technologically or mathematically sophisticated, the downside of many such deep learning methods is a sacrifice of interpretability.

Our paper instead takes on a network-based approach to prediction. A few papers have considered such methodologies, such as [9]–[11]. Most of these methods use standard graph metrics such as in-degree, out-degree, or a "baseline" PageRank approach. In this paper, we refine the PageRank approach [1] and propose an alternative scheme that factors in point-differential and team records when constructing the sports network graphs.

B. Paper Organization and Contributions

In Section II, we present our network-driven prediction methodology, starting with the directed graph of sports teams and the baseline PageRank algorithm. We then propose a novel weighting scheme in which we modulate the weights of the graph according to game point differentials.

Following this, we evaluate both methods in Section III, where we consider their ability to predict the outcomes of games when trained on a certain portion of a sports season. We first consider data from National Football League (NFL) seasons between 2007 and 2017, in which we find that our weighted method outperforms the baseline by 2.5% on average, and obtains up to 70% accuracy in predicting the outcomes of games. We then run some preliminary analysis on National Basketball Association (NBA) data using just the weighted method.

Lastly, we describe some improvements and modifications to our method in Section IV, and conclude in Section V while providing some possible future directions.

II. Network Model Specification

We consider the network of sports teams as a weighted directed graph. This has similarities to the PageRank model of the web [1], in which webpages are visualized as nodes in a graph, and links between pages as edges. In its simplest form, the ranking of a webpage is determined by the traffic generated to that page by a random surfer on the graph.

978-1-5386-0579-0/18/$31.00 ©2018 IEEE

TABLE I
TOY EXAMPLE NFL DATA THAT WE USE TO DEMONSTRATE THE TEAMRANK METHODS.

Winner	Loser	Winning Score	Losing Score	Differential
Broncos	Panthers	21	20	1
Broncos	Saints	25	23	2
Cowboys	Redskins	27	23	4
Saints	Panthers	41	38	3
Cowboys	Redskins	31	26	5
Panthers	Saints	23	20	3

TABLE II
TEAM INDICES FOR THE EXAMPLE IN TABLE I.

Team	Index
Broncos	1
Panthers	2
Saints	3
Cowboys	4
Redskins	5

In our case, the teams in our league of interest represent nodes in a graph $G = (V, E)$. The edges E between nodes V in G encode game results. A team is connected to another team if and only if it has played a game against them. The direction of each link encodes the winner/loser relationship of a game. A node $i \in V$ points to another node $j \in V$ if team i lost to team j in a game (ties excluded). The link weights $A_{ij} \in E$ differ between our (1) "baseline TeamRank" and (2) "weighted TeamRank" method.

Baseline TeamRank. In the baseline TeamRank method (Section II-A), we consider only the win/loss outcome of games. Here, we set the link weight A_{ij} from node i to j as 1 if the link exists and 0 if the link does not.

Weighted TeamRank. Under the weighted TeamRank method (Section II-B), the link weights A_{ij} are the point differentials (i.e. by how many points team i lost to team j). In this manner, we hope to capture the effect of teams winning by larger margins. We would expect, holding other factors equal, that teams with higher margins of victory are stronger.

It is possible to have repeated games in a season. In this scenario, under the baseline case, we ignore repeated-outcome games. Under the weighted case, however, we take the sum of the point differentials as the weighting on the link.

In what follows, we will formalize the two methods. In doing so, we will use a toy example of five NFL teams playing in a league. The teams play a total of five games, as summarized in Tables I. For our calculations, we assign each team a unique index as in Table II.

A. Baseline TeamRank Method

In the baseline method, link weights are determined as

$$A_{ij} = \begin{cases} 1 & \text{team } i \text{ lost to team } j \\ 0 & \text{otherwise} \end{cases}$$

This gives us the $N \times N$ adjacency matrix A_{base} visualized in Fig. 1 where N is the number of teams. From this, we compute the H matrix (row-normalized version of the A matrix) as detailed in the PageRank algorithm:

$$A_{base} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$H_{base} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

There are two all-zero rows in the above H matrix. This is a result of dangling nodes, or nodes that represent undefeated teams in this sports team graph. In the PageRank algorithm, dangling nodes are dealt with by adding directed links of equal weight from the dangling node to each of the other nodes. This gives us a row-normalized stochastic matrix $\hat{H}$ with $\sum_j \hat{H}_{ij} = 1$ for all i. In this example, we get

$$\hat{H}_{base} = \begin{pmatrix} 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.5 & 0 & 0.5 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

In our network, this method of row-normalizing the adjacency matrix is tantamount to adding a small probability of defeat of the undefeated team by each of the other teams. In Section IV-B, we reexamine the consequences of this approximation and try different alterations to this dangling node rule.

In the PageRank algorithm, a randomization parameter θ is added to ensure that the stochastic matrix describes a random walk on a connected graph. This gives us a stochastic matrix P:

$$P \equiv \theta \hat{H} + (1-\theta)\frac{1}{N}\mathbf{1}\mathbf{1}^T$$

where $\mathbf{1}$ is the vector of all ones. Setting $\theta < 1$ ensures that our network does not consist of disconnected cliques of teams that have not played each other.

For our toy example, we get the following P matrix when we use $\theta = 0.85$:

$$P_{base} = \begin{pmatrix} 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.455 & 0.03 & 0.455 & 0.03 & 0.03 \\ 0.455 & 0.455 & 0.03 & 0.03 & 0.03 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.03 & 0.03 & 0.03 & 0.88 & 0.03 \end{pmatrix}$$

We will use this value of θ for the evaluation in Sec. III, and explore its effect further in Sec. IV.

The importance vector π^* we use for ranking the teams is the normalized left eigenvector of P corresponding to its largest eigenvalue. It is the stationary distribution corresponding to a Markov chain described by the transition matrix P and may be obtained using the power method as

$$\pi^{*T} = \lim_{k\to\infty} \pi^T[k] = \lim_{k\to\infty} \pi^T[0]P^k,$$

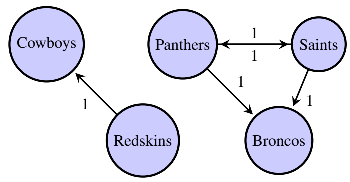

Fig. 1. Adjacency graph A_{base} for our toy example.

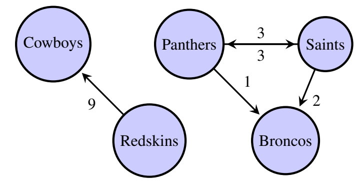

Fig. 2. Adjacency matrix A_{weight} for our toy example.

TABLE III
RESULTS FROM THE BASELINE TEAMRANK METHOD IN THE TOY EXAMPLE.

Ranking	Team	Record	π^*
1	Broncos	2-0	.2814
2	Cowboys	1-0	.2101
3	Saints	1-2	.1975
4	Panthers	1-2	.1975
5	Redskins	0-1	.1136

TABLE IV
RESULTS FROM THE WEIGHTED TEAMRANK METHOD IN THE TOY EXAMPLE.

Ranking	Team	Record	π^*
1	Saints	1-2	.2472
2	Broncos	2-0	.2344
3	Panthers	1-2	.2284
4	Cowboys	1-0	.1885
5	Redskins	0-1	.1019

where $\pi^T[0] = \left(\frac{1}{N} \quad \frac{1}{N} \quad \cdots \quad \frac{1}{N}\right)$. We will use this method to determine our predictions and rankings in Section III.

After $k = 20$ iterations for our toy example, we obtain the π^* vector given in Table III. These rankings are consistent with the records of the teams.

B. Weighted TeamRank Method

Now we explore the weighted method, which differs from the baseline in how the initial adjacency matrix is calculated:

$$A_{ij} = \begin{cases} w_{ij} & \text{team } i \text{ lost to team } j \\ 0 & \text{otherwise} \end{cases}$$

where w_{ij} is the point differential, i.e., the total number of points team i lost by to team j. This means that if team j beat team i twice in a season, A_{ij} is the sum of the two point differentials in those matches. In the sports network graph, the idea is that more "traffic" will be allocated to the teams with the most significant victories. For example, suppose teams i and j have the same win/loss record, but while team i has beaten several bad teams by small margins and lost to good teams by large margins, team j has had the opposite fortune. Objectively speaking, team j is stronger than team i although this is not evinced by their equivalent records. Weighted PageRank accounts for this by factoring in point differential into the network construction.

The resulting adjacency matrix A_{weight} for our toy example is visualized in Fig. 2. After determining the stochastic matrix P, we apply the power method again, and obtain the set of rankings in Table IV.

Unlike the baseline TeamRank scheme, the weighted method suggests that the Saints are stronger than the Broncos in our toy example. This finding may be contradictory to our intuition based on the win/loss record, but it is likely that this is due to the smallness of the dataset and an artifact of the specific toy example chosen. In the next section, we use the above PageRank schemes on real data and demonstrate their predictive power as well as the superiority of the weighted scheme to the baseline scheme.

III. RESULTS AND DISCUSSION

We predict outcomes of NFL (Sec. III-A) and NBA (Sec. III-B) matches in a season using historical data from the same season. For both the baseline and weighted TeamRank algorithms, we predict team i will beat team j if $\pi^*[i] > \pi^*[j]$, i.e., if its importance eigenvector is larger.

A. NFL Predictions

For our NFL predictions, we take the 17 weeks of each season and divided them into a series of training windows (weeks of data on which we trained the network) and testing windows (subsequent weeks on which we evaluated the predictions). We used training windows ranging from the first three weeks up to the first 12, and used these to predict results for the rest of the season, through the postseason (playoffs and superbowl) when available. Overall, we find that while both methods have predictive value, the weighted method outperforms the baseline.

1) Dataset: The dataset we used was from Pro Football Reference (www.pro-football-reference.com). In Fig. 3, we visualize the the 2016 NFL Season network trained on the first 12 weeks in the weighted TeamRank scheme.

2) Prediction Results: In Fig. 4, we plot the prediction accuracy throughout the 2016-17 NFL season (the most recent, complete season) for both TeamRank schemes. In Table V, we report the prediction accuracies, season by season, of the two schemes when we use 12 weeks of data to train. In Fig. 4, we see that both methods tend to improve with more training weeks, and that the weighted method outperforms the baseline in each case, reaching up to 70% accuracy. Table V further shows superiority of the weighted scheme, which either matches or outperforms the baseline in 8 of the 11 seasons tested, obtaining about a 5% improvement in accuracy on average. For brevity in the rest of the paper, we will proceed with the weighted scheme only.

3) 2017-18 Team Rankings: Apart from simply predicting match outcomes, we also want to develop power rankings of the teams in the league. The TeamRank methods give a

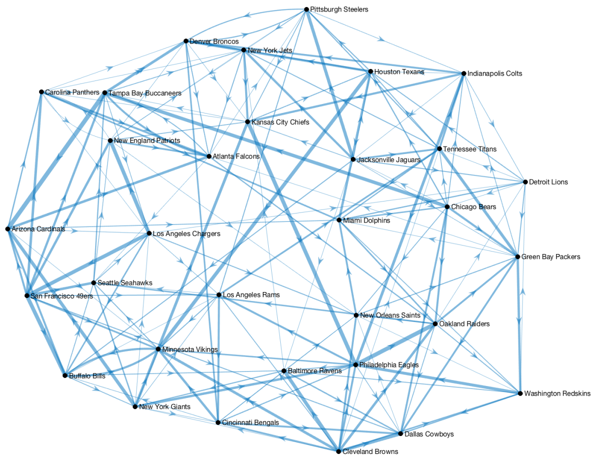

Fig. 3. Visualization of the weighted TeamRank graph for the 2016-17 NFL Season, taking weeks 1-12. Node labels are the teams, while edge thickness is weighted by point differentials.

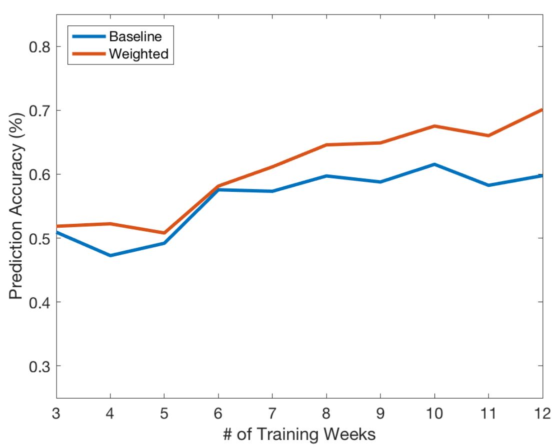

Fig. 4. Prediction accuracy of the baseline and weighted TeamRank algorithms for the 2016-17 NFL season as the number of weeks in the training window is varied. Performance improves with more training data, with the weighted method reaching roughly 70% by week 12.

straightforward way of doing so – we can sort the teams by their importance in the π^* vector.

In Table VI, we give rankings of the top 16 NFL teams in the 2017-18 season as of 1/24/18 along with each team's current record. The results of our weighted power ranking generally match the ranking of teams by win/loss record, but also provide some variation depending on the teams' strengths of victories (i.e., which teams they have beaten). For example, though the Chiefs and Jaguars have lower records than the Patriots and Steelers, the Chiefs beat the Patriots and the Jaguars beat the Steelers. Also note that the NFL is equally divided into two conferences: the NFC and AFC, each containing 16 teams. It is interesting to note here that the first four teams in this list all come from the AFC conference even though the algorithm incorporates cross-conference games equally. This means that our method potentially has the ability

TABLE V
PREDICTION ACCURACY OF THE TEAMRANK METHODS ACROSS DIFFERENT NFL SEASONS, WITH 12 WEEKS OF TRAINING DATA. THE WEIGHTED METHOD OUTPERFORMS THE BASELINE IN 8 OF 11 SEASONS.

Season	Baseline	Weighted
2007	**0.659**	0.602
2008	0.489	**0.511**
2009	0.580	0.580
2010	0.580	**0.602**
2011	0.636	**0.648**
2012	0.625	**0.670**
2013	**0.625**	0.602
2014	0.568	**0.648**
2015	**0.636**	0.625
2016	0.598	**0.701**
2017	0.591	**0.682**
Avg	0.5988	**0.6246**

TABLE VI
RESULTS OF USING THE WEIGHTED TEAMRANK METHOD TO RANK THE NFL TEAMS IN THE 2017-18 SEASON. THE TOP FOUR TEAMS ARE ALL IN THE AFC CONFERENCE.

Team	Rank	Wins	Losses
Kansas City Chiefs	0.0711	10	7
Jacksonville Jaguars	0.0678	12	6
New England Patriots	0.0598	14	3
Pittsburgh Steelers	0.0585	13	4
Minnesota Vikings	0.0539	14	3
Los Angeles Rams	0.0526	11	6
New Orleans Saints	0.0471	12	6
Philadelphia Eagles	0.0455	14	3
Dallas Cowboys	0.0434	9	7
Seattle Seahawks	0.0400	9	7
Tennessee Titans	0.0400	10	8
Atlanta Falcons	0.0370	11	7
Carolina Panthers	0.0357	11	6
San Francisco 49ers	0.0325	6	10
Baltimore Ravens	0.0270	9	7
Miami Dolphins	0.0267	6	10

to recognize a stronger conference within the league.

B. NBA Results

We also applied our method to predictions for the NBA. Here, we cannot train on a weekly basis since in the NBA, a team may play on consecutive days or barely play in a week at all. Therefore, we instead resorted to monthly training windows. In particular, we considered training windows of 1-5 months in size to construct the adjacency matrix, and then predicted the rest of the regular season based on the importance vector scores π^*.

1) Prediction Results: In Table VII, we summarize the prediction accuracies obtained by the weighted PageRank method as a function of training window size. These results are lower quality than what was obtained for the NFL, but have up to 18% improvement over a random predictor. It is possible that lower quality predictions in the NBA are due to (i) the teams in the NBA tending to be closer in ability and competitiveness, and (ii) outcomes of games tending to be more volatile due to their higher frequencies.

2) 2016-17 Team Rankings: We also compiled rankings of all 30 NBA teams using the five month training window. In Table VIII, we include the top 10 teams generated using the entire 2016-17 NBA season. As with the NFL, we see an interesting clustering by league conferences. The NBA is equally divided into Eastern and Western Conferences, and

TABLE VII
WEIGHTED TEAMRANK PREDICTION RESULTS APPLIED TO THE NBA FOR DIFFERENT TRAINING WINDOWS.

Months Trained	Accuracy
1	0.5468
2	0.5563
3	0.5835
4	0.5961
5	0.5602

TABLE VIII
2016-17 TEAM RANKINGS OF THE NBA PRODUCED BY THE WEIGHTED TEAMRANK SCHEME. THERE IS A SIMILAR CLUSTERING BY LEAGUE CONFERENCES AS WITH THE NFL.

Team	Rank	Wins	Losses
Golden State Warriors	0.0778	62	14
San Antonio Spurs	0.0657	58	17
Los Angeles Clippers	0.0531	46	31
Memphis Grizzlies	0.0493	42	34
Houston Rockets	0.0457	51	25
Utah Jazz	0.0421	47	29
Denver Nuggets	0.0393	35	40
Cleveland Cavaliers	0.0370	48	27
Toronto Raptors	0.0366	46	30
Oklahoma City Thunder	0.0362	43	32

the top 7 teams in this ranking are all from the Western Conference, with the eventual runner-up Cleveland Cavaliers (from the Eastern Conference) taking the 8th spot.

IV. ALGORITHMIC EXTENSIONS

Now that we have evaluated the weighted method, we explore several possible extensions that may achieve further improvements in accuracy, including the handling of dangling nodes (Sec. IV-B) and a recency bias for team momentum (Sec. IV-C). We consider these with the NFL datasets.

A. Parameter Tuning

As noted in Section II, we set the randomization parameter $\theta = 0.85$, as is standard. In Table IX, we give the results from varying θ for the weighted TeamRank algorithm on the 2012-16 seasons. While $\theta = 0.9$ gives the on-average best results, the differences observed for $\theta \in [0.65, 0.99]$ are slight. This justifies the choice of $\theta = 0.85$.

B. Dangling Nodes and Self-Weighting Scheme

Recall from Section II that we created the $\hat{H}$ matrix by replacing loss-rows of undefeated teams with uniform distributions across all other teams. However, an undefeated team is not equally likely to lose to all the other teams. One possible solution to this problem is including self-links in the adjacency matrix and weighting them by each team's win percentage. Although this self-weighting scheme is inspired by a desire to improve the description of undefeated teams, we can use it for all teams in the row-normalized transition matrix $\hat{H}^{(sw)}$.

To do this, we first run the normal weighted method described in Section II-B to calculate the row-normalized adjacency matrix $\hat{H}$. Then, for any team i in the adjacency matrix, we set the probability of a self loop equal to the win percentage of that team. Namely, we let:

$$\hat{H}^{(sw)}_{(i,i)} = \frac{\text{\# of games won by Team } i}{\text{\# of games played by Team } i}$$

TABLE IX
PREDICTION ACCURACY VARYING θ. $\theta = 0.9$ SHOWS THE BEST RESULT, BUT THE DIFFERENCES ARE SLIGHT OVERALL.

Year	θ 0.65	0.7	0.75	0.8	0.85	0.9	0.95	0.99
2012	0.659	0.659	0.659	0.659	0.67	0.681	0.647	0.647
2013	0.602	0.59	0.59	0.59	0.602	0.602	0.59	0.579
2014	0.613	0.613	0.613	0.625	0.647	0.659	0.659	0.659
2015	0.67	0.647	0.647	0.636	0.625	0.625	0.625	0.625
2016	0.689	0.689	0.689	0.701	0.701	0.701	0.701	0.701
Average	0.647	0.640	0.640	0.642	0.650	**0.654**	0.644	0.642

TABLE X
PREDICTION ACCURACY OF THE SELF-WEIGHTED TEAMRANK METHOD VERSUS THE PREVIOUS BASELINE AND WEIGHTED METHODS. THE WEIGHTED METHOD PERFORMS THE BEST ON AVERAGE, BUT THE SELF-WEIGHTED GIVES THE HIGHEST SINGLE-SEASON ACCURACY.

Season	Baseline	Weighted	Self-Weighted
2007	**0.659**	0.602	0.614
2008	0.489	0.511	**0.5682**
2009	0.580	0.580	**0.614**
2010	0.580	**0.602**	0.591
2011	0.636	**0.648**	0.591
2012	0.625	**0.670**	0.648
2013	**0.625**	0.602	0.614
2014	0.568	**0.648**	0.591
2015	**0.636**	0.625	0.614
2016	0.598	**0.701**	0.678
2017	0.591	0.682	**0.705**
Avg	0.5988	**0.6246**	0.621

The off-diagonal terms $H^{(sw)}_{ij}$ for $j \neq i$ are calculated by the following rule:

$$\hat{H}^{(sw)}_{(i,j)} = \frac{\text{\# of games lost by Team } i}{\text{\# of games played by Team } i} \times \hat{H}_{(i,j)}$$

The interpretation of this scheme is as follows. A team will transition back to itself with probability equal to its winning percentage. If the team does not transition to its own node, it transitions to another node with the old $\hat{H}_{ij}$ probability as shown above. Recall that the original $\hat{H}_{ij}$ matrix was constructed using point differentials; this new method, therefore, takes into account both point differentials and win percentages.

The rest of the procedure, including the calculations of the stochastic matrix P and the importance vector π^* is as in Section II-B. The prediction accuracy results are shown in Table X compared with the original baseline and weighted TeamRank algorithms. Although the weighted scheme performs the best on average, the self-weighted method is not far behind and actually yields the highest single season prediction probability of roughly 71% in the 2017 season.

The power rankings results for this method are given in Table XI, for the top half of the teams in the 2017 NFL season. This set of results has some additional insights over the corresponding ones for the Weighted TeamRank method in Table VI. First, it more closely matches the ranking of the teams by record. Second, the top 3 teams (Eagles, Patriots and Vikings) are all teams that competed in the NFL Conference Championships (semifinals).

C. Recency Bias for Momentum

Momentum can be a significant factor driving the performance of a team, particularly towards the end of the season.

TABLE XI
RESULTS OF USING THE SELF-WEIGHTING TEAMRANK METHOD TO RANK TEAMS IN THE 2017-18 SEASON. COMPARED WITH TABLE VI, NOW THE TOP THREE ALL COMPETED IN THE CONFERENCE CHAMPIONSHIPS.

Team	Rank	Wins	Losses
Philadelphia Eagles	0.0726	14	3
New England Patriots	0.0672	14	3
Minnesota Vikings	0.0612	14	3
Los Angeles Rams	0.0581	11	6
New Orleans Saints	0.0577	12	6
Pittsburgh Steelers	0.0574	13	4
Jacksonville Jaguars	0.0560	12	6
Kansas City Chiefs	0.0438	10	7
Carolina Panthers	0.0408	11	6
Tennessee Titans	0.0385	10	8
Baltimore Ravens	0.0382	9	7
Atlanta Falcons	0.0356	11	7
Dallas Cowboys	0.0321	9	7
Seattle Seahawks	0.0321	9	7
Buffalo Bills	0.0311	9	8
Detroit Lions	0.0269	9	7

TABLE XII
RESULTS IN PREDICTING THE OUTCOMES OF GAMES IN WEEK 13 OF NFL SEASONS, FOR DIFFERENT COMBINATIONS OF WEIGHTED (WTD), SELF-WEIGHTING (SWTG), AND RECENCY.

Season	Wtd	Wtd+Recency	SWtg.	Swtg+Recency
2007	0.5625	0.625	0.562	0.75
2008	0.4375	0.5625	0.437	0.625
2009	0.625	0.5625	0.562	0.625
2010	0.8125	0.8125	0.812	0.75
2011	0.6875	0.6875	0.687	0.625
2012	0.625	0.625	0.562	0.625
2013	0.5	0.5	0.437	0.562
2014	0.5625	0.5	0.5	0.562
2015	0.6875	0.5625	0.687	0.562
2016	0.5333	**0.8**	0.533	**0.8**
2017	0.6875	0.6875	0.687	0.75
Avg	0.611	**0.630**	0.588	**0.657**

To investigate this, we include an additional parameter λ in the weighted TeamRank method that serves as a momentum factor. In particular, we adjust the edge weights in the adjacency matrix according to $A_{ij} = \lambda^t w_{ij}$, where t is number of weeks elapsed since the match in consideration.

In Table XII, we show the results of predicting outcomes in Week 13 of the NFL seasons for different combinations of the weighted or self-weighting schemes with or without the recency bias, training over Weeks 1-12 and taking $\lambda = 0.95$. Predicting games for a single week is different from what we have done previously in predicting all games after a certain week, but more closely captures the effect of momentum shifts. Overall, we see that recency can improve predictions for both schemes: in particular, both of combinations of recency obtain 80% in the 2016 season, and we obtain an average accuracy of almost 66% when self-weighting is combined with recency.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed and implemented network-driven algorithms for predicting the outcomes of professional sports games, considering both the NFL and NBA. Formulating directed graphs of sports teams based on the outcomes of individual games, our baseline TeamRank method obtained prediction accuracies on NFL games of up to 60%, while our weighted TeamRank method pushed this to 70% while yielding analytics on team rankings as well. We also considered variations of our weighted method to account for team win records and shifts in momentum in terms of how the directed graph of teams is defined. Overall, our results demonstrated the validity of this network-driven approach, and provided a more interpretable and intuitive alternative to standard machine-learning based sports analytics methods.

There are several avenues of future work that we are exploring. First is week-by-week predictions: it would be interesting to create a prediction algorithms that predicts the outcomes of a specific game of interest using all data available up through the present time. This is computationally intensive as it requires a new run of TeamRank after every week for the NFL and an update every day for the NBA. Second is home-field advantage: teams notoriously have better records at home than on the road, so factors could be incorporated to model the fact that a team playing on its home field will have a more supportive crowd and be more familiar with the field. Third is rivalry games: these could be thought of as having more "weight" than ordinary games, given that the fans are more intense, and the teams have more pride on the line. It is known in the NFL at least, that when intra-division games happen, records are "off-the-table." Fourth is extensions to other sports: while we have applied our analysis and algorithms to the NFL and NBA, in the future, we want to do a complete analysis of other sports including the NBA for several seasons. We believe these network-driven approaches could also provide value to analytics for the National Hockey League (NHL) and Major League Baseball (MLB), for example.

REFERENCES

[1] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer Networks and ISDN Systems*, vol. 30, no. 1, pp. 107 – 117, 1998, proceedings of the Seventh International World Wide Web Conference.

[2] K. Goldsberry, "Courtvision: New visual and spatial analytics for the nba."

[3] K. M. Kniffin, T. Howley, and C. Bardreau, "Putting muscle into sports analytics: Strength, conditioning, and ice hockey performance," *The Journal of Strength & Conditioning Research*, vol. 31, no. 12, pp. 3253–3259, 2017.

[4] G. Baio and M. Blangiardo, "Bayesian hierarchical model for the prediction of football results," *Journal of Applied Statistics*, vol. 37, no. 2, pp. 253–264, 2010.

[5] D. Delen, D. Cogdell, and N. Kasap, "A comparative analysis of data mining methods in predicting ncaa bowl outcomes," *International Journal of Forecasting*, vol. 28, no. 2, pp. 543 – 552, 2012.

[6] A. Joseph, N. Fenton, and M. Neil, "Predicting football results using bayesian nets and other machine learning techniques," *Knowledge-Based Systems*, vol. 19, no. 7, pp. 544 – 553, 2006, creative Systems.

[7] R. Shah and R. Romijnders, "Applying deep learning to basketball trajectories," *arXiv preprint arXiv:1608.03793*, 2016.

[8] K.-C. Wang and R. Zemel, "Classifying nba offensive plays using neural networks," in *Proceedings of MIT Sloan Sports Analytics Conference*, 2016.

[9] A. Y. Govan, C. D. Meyer, and R. Albright, "Generalizing google's pagerank to rank national football league teams," 2008.

[10] A. Y. Govan, A. N. Langville, and C. D. Meyer, "Offense-defense approach to ranking team sports," *Journal of Quantitative Analysis in Sports*, vol. 5, no. 1.

[11] V. Lazova and L. Basnarkov, "Pagerank approach to ranking national football teams," *arXiv preprint arXiv:1503.01331*, 2015.